A GAME OF MUSICAL LAYERS:

AN INVESTIGATION INTO THE USE OF A VERTICAL LAYERING SOUNDTRACK AS AN INTEGRAL PART OF THE INFORMATION SYSTEM OF A VIDEO GAME

WILL ASHTON

A submission presented in partial fulfilment of the
requirements of the University of South Wales
for the degree of Master of Philosophy

January 2015

Contents

Abstract........iv

Definition of Vertical Layering........v

List of Figures........vi

List of Tables........ix

List of DVD Items........x

Acknowledgements........xi

1 Introduction........1

2 Game Audio Theory........5

Hearing and Listening........6

The Diegesis........9

Actions and Interactions........14

Structures........18

Information Systems........22

Usability Function........30

3 Audio in Real-time Strategy Games........33

Gameplay........34

Musical Soundtrack........41

4 Methodology........52

5 Level 1: A Basic Vertical Layering Soundtrack........59

Design........59

Results........73

Feedback....82
Conclusion....85
6 Level 2: Vertical Layering with On/Off States....89
Design....89
Results....100
Feedback....110
Conclusion....115
7 Level 3: Multi-dimensional Information....119
Design....119
Results....133
Feedback....157
Conclusion....161
8 Conclusion....165
Findings....165
Reflections and Future Directions....171

Appendices
Appendix 1: Level 1 Tutorial Transcript....175
Appendix 2: Level 2 Tutorial Transcript....177
Appendix 3: Level 3 Tutorial Transcript....180

Bibliography....184

Abstract

This thesis contributes to the field of game audio research by investigating the hypothesis that a vertical layering musical soundtrack can be used as an integral part of the information system of a video game. The concept of dynamically controlling music through multiple layers, as well as the idea that music can be used to convey information to players, are currently underexplored areas within game audio research.

The analytical and information-rich genre of real-time strategy games is used to contextualise the research, although this research has wider relevance to other genres, as well as the field of game audio in general. The work presented here is supported by a discussion of the relevant knowledge base and an analysis of the audio in a selection of current real-time strategy games.

In order to test the hypothesis, and highlight the strengths and weaknesses of the approach, this research employed a methodology that consisted of empirical user testing of three purpose-built game levels, featuring vertical layering soundtracks and in-built testing mechanics, in order to produce both quantitative results and qualitative feedback from a small group of research participants about the effectiveness of the information transfer achieved by these vertical layering soundtracks.

As was shown throughout all three test levels, players appeared to be able to learn individual layers of a soundtrack, recognise them during gameplay and respond accordingly to the information they conveyed, while attending to other gameplay tasks. Based on the findings of the research, potential design considerations for the use of vertical layering soundtracks as a source of information have been put forward which could be used as a basis for further research within the topic.

which stacks these layers on top of one another in perfect synchronization. Interactivity is achieved through the independent manipulation of the layers, enabling the overall track to change in accordance with the fluctuating state of the game." (Phillips, 2014: 194)

List of Figures

3.1 Screenshot of *StarCraft II*...34

3.2 *StarCraft II* worker and mineral resource...................................35

3.3 A hero from *Strife* using a special ability....................................36

3.4 A hero portrait from *Dota 2*..38

3.5 Fog of war in *Dota 2*..41

3.6 Layout of *Strife* level...46

5.1 Flame turrets defending the player's base in Level 1.......................61

5.2 Start screen of Level 1..62

5.3 Extract from Level 1 tutorial..63

5.4 Level 1 prediction buttons..65

5.5 Level 1 results screen...66

5.6 Level 1: extract from base layer..67

5.7 Level 1: extract from layer 1..68

5.8 Level 1: extract from layer 2..68

5.9 Level 1: extract from layer 3..69

5.10 Player performance over time in Level 1.....................................80

5.11 Gameplay/musical information mismatch in Level 1.........................86

6.1 Micro-task in Level 2...91

6.2 Level 2: extract from base layer..95

6.3 Level 2: extract from layer 1..95

6.4 Level 2: extract from layer 2..95

6.5 Level 2: extract from layer 3..96

6.6 Level 2: extract from layer 4...97

6.7 Level 2: extract from layer 5...97

6.8 Original Fmod event set-up in Level 2...98

6.9 Correct predictions per layer in Level 2...102

6.10 Correct predictions per layer in Level 2, including "one-way"...103

6.11 Player performance over time in Level 2...106

6.12 Prediction performance and micro-task in Level 2...107

6.13 First wave and overall performance in Level 2...109

7.1 Macro-task of Level 3...120

7.2 The three enemy types of Level 3...122

7.3 Prediction toggle buttons for Level 3...123

7.4 Level 3 tutorial music example buttons...125

7.5 New models for Level 3...126

7.6 Extract from the yellow enemy layer in Level 3...127

7.7 Extract from the red enemy layer in Level 3...128

7.8 Extract from the brown enemy layer in Level 3...128

7.9 Extract from the base layer in Level 3...129

7.10 Extract from the calm layer in Level 3...130

7.11 Extract from the danger layer in Level 3...130

7.12 Volume automation for Level 3 motivic layers...132

7.13 Overall prediction success rate by category in Level 3...138

7.14 Predictions for small groups of enemies in Level 3...140

7.15 Predictions for large groups of enemies in Level 3...141

7.16 Predictions for absent groups of enemies in Level 3...142

7.17 Player performance over time in Level 3...143

7.18 Micro- and macro-tasks in Level 3....................................145

7.19 Micro-task and overall prediction success in Level 3..............146

7.20 Macro-task and overall prediction success in Level 3..............147

7.21 Success of individual motivic layers in Level 3......................148

7.22 Enemy type prediction success by group size in Level 3...........150

7.23 a–c Predictions for yellow enemy groups in Level 3...............153

7.24 a–c Predictions for red enemy groups in Level 3....................154

7.25 a–c Predictions for brown enemy groups in Level 3................155

List of Tables

5.1 Each player's results for Level 1 74

5.2 Total correct predictions per layer in Level 1 75

5.3 Confusion between layers in Level 1 77

5.4 Potential wave-type influence in Level 1 81

6.1 Each player's results for Level 2 101

6.2 Confusion between layers in Level 2 104

6.3 Total number of predictions for each layer in Level 2 105

7.1 Each player's results for Level 3 134

7.2 Comparison of auto-accepted results in Level 3 135

7.3 Results for the micro- and macro-tasks in Level 3 136

List of DVD Items

00) Readme

01) Level 1: all layers
02) Level 1: base layer
03) Level 1: layer 1
04) Level 1: layer 2
05) Level 1: layer 3
06) Level 1: video compilation

07) Level 2: all layers
08) Level 2: base layer
09) Level 2: layer 1
10) Level 2: layer 2
11) Level 2: layer 3
12) Level 2: layer 4
13) Level 2: layer 5
14) Level 2: video compilation

15) Level 3: all layers
16) Level 3: yellow layer
17) Level 3: red layer
18) Level 3: brown layer
19) Level 3: base layer
20) Level 3: calm layer
21) Level 3: danger layer
22) Level 3: video compilation

Level 1: test executable
Level 2: test executable
Level 3: test executable

Acknowledgements

I would like to thank my Director of Studies, Paul Carr, and supervisor, Robert Smith, for being kind enough to take this project on at a late stage and for helping me over the finishing line with their support and advice in the final months of the research.

I would also like to thank all of the test subjects who participated in my work, particularly those who had the stamina to complete all three game test levels, and whose feedback helped to shape the direction of this research.

I am very grateful to sound designers Roland Shaw of Valve and Stephen Baker of S2 Games for being kind enough to take the time to answer the various questions I had about sound and music in their fantastic games.

Finally, my sincerest thanks go to my original supervisor and mentor, Ben Challis, for all his help, encouragement and enthusiasm, and for all the tutorials and countless Skype conversations that contributed so much to this research. I am extremely grateful for his continued support.

1

Introduction

> *Gaming is the most vital artform of the age ... a field that has seen and continues to see an enormous ferment of creativity, a field that may well become the predominant artform of the 21st century, as film was of the 20th, as the novel was of the 19th. By God, we're privileged to be here at the birth of this great form, of the creation of a democratic artform for a democratic age, the creation of structures of desire, of ways to enable people to create their own entertainment through play.*
>
> (Costikyan, 2001: 8)

Video game music has undergone a continuous transformation over the past forty years since the primitive sounds of early arcade machines, home computers and consoles. Game music now reflects the sheer variety of styles and genres that exist in other musical contexts. Music has become established as a vital part of the modern-day gaming experience, not only enhancing but in many cases being a crucial part of the actual gameplay of many games. Once, game composers could only dream of matching the aesthetic qualities of music in other, more established, entertainment media such as film, but now game music can arguably not only equal the artistic qualities of film music, but create a truly immersive, dynamic and even interactive audiovisual experience for the player.

Equally, the academic study of video game music has developed rapidly since the turn of the century: from early pieces championing the legitimacy of video game music both as big business and an art form (see, for example, Belinkie, 1999), up to more recent work, looking at the functionality and structures of contemporary dynamic game music (see, for example, Collins, 2008a). The research put forward here aims to continue this direction of exploration. While much has been written about how music can have an emotional impact, reinforce narrative and characters and affect players by immersing them in a fictional world, this research aims to explore a less well understood aspect of music in games – its ability to communicate *information* to players.

Music in a dynamic and interactive medium like video games can be considered a part of the interface between a player and a game, helping to create a more comprehensible and usable system. This function of music will be explored through one particular type of musical composition and structure used within games, often referred to as *vertical layering*. This type of music is composed and controlled in individual parts, or layers, that can be added and subtracted from the soundtrack overall. This dynamic manipulation of music during gameplay therefore has huge potential to communicate information elegantly to a player about virtually any aspect of the gameplay experience.

Games themselves are vastly complex information systems: every object, mechanic and interaction a player experiences is rich with information. Understanding all that the game communicates to a player is a vital part of interacting with a game, but even more vital to overcoming the many obstacles and challenges that designers put in front of their players. While understanding information plays an important role in all games, the niche genre of real-time strategy (RTS) games is arguably one that puts considerable emphasis on this aspect of video games, and reading and interpreting gameplay information is a core part of their design.

This research will therefore use the information-rich genre of RTS games as a context in which to explore the ability to convey gameplay information in vertical layering music. In order to do this, a series of three purpose-built RTS-style game levels were developed using the game engine Unity3D in order to explore and test the ability of music to convey information to a player. This method of research has only recently become feasible given the advent of more accessible game engines, such as Unity3D, which has led to the democratisation of game development.

Few people would argue that music cannot create mood and atmosphere within a game. But if music does not directly follow the dynamic flow of gameplay, then it does not communicate anything pertinent to the particular situation that a player might find themselves in. Music like this that does not support the game design and direction specifically can potentially become irrelevant and even distracting for the player. Particularly in games that are especially difficult or that contain a competitive element, any feature that does not directly support the player's ability to succeed in their goals will be deemed unnecessary. This has led to a culture of some players turning the music off or even replacing it with their own. It should therefore be every designer's and composer's goal to create music that supports the mood, emotion and atmosphere of the game, but that is also integrated into its usability system by making sure the music dynamically supports and conveys information about the events and actions within the game. In this way, a game's soundtrack will not be a needless distraction but add even more relevance and value to the experience of the player.

By studying how music functions in video games, and by looking at the information-rich context of RTS games in particular, this research aims to investigate how a vertical layering soundtrack can be used as an integral part of the information system of a video game.

2

Game Audio Theory

> *Music has the potential to be more than window dressing for games. It can directly support gameplay and heighten emotion. With a healthy collaboration between game designer and composer, music can be an effective game design tool, helping to establish and reinforce the core game design.* (Whitmore, 2003: n.p.)

The knowledge base on video game audio, while still a relatively new field of study, is ever expanding, and many authors are now discussing game audio, not only in its emotional and affecting role, but as it relates to communicating gameplay relevant information to players. Sound design and music often overlap in their aural characteristics, and all aspects of audio in games are interrelated and share many functions and goals. Therefore, this chapter will draw from literature that discusses all types of game audio, including sound effects and speech, as well as music specifically, as they relate to conveying information to a player.

In many ways, game audio has been defined by the technological development of gaming hardware (see Weinen, 2007: 3). In the early days of arcade machines and home consoles, audio was very limited by the technology of the time (see Belinkie, 1999; Leonard, 2001). Despite this, game audio has always had a functional aspect that aided player interaction. Even *Pong*'s (Atari, 1972) simple "blip" informed the players that the ball had hit a paddle, and *Space Invaders*' (Midway, 1978) proto-musical, steadily increasing pulse cued the player to respond to the ever-rising threat of the oncoming aliens (see Collins, 2005: 2; Jørgensen, 2009: 18–19).

While many theorists have studied games and game music in the context of narrative and story-telling, particularly drawing from the field of film studies, arguably not enough attention has been given in the literature to understanding game music from the perspective of game design. Video games are interactive systems and therefore the position of the player as an active *participant*, and not just a passive *viewer*, is the reason that sound and music take on a functional role in the game's information system. Salen and Zimmerman (2004: ch. 8, p. 7) identify four traits that define all games, but particularly apply to video games: "Immediate but narrow interactivity; manipulation of information; automated complex systems; [and] networked communication". These aspects should therefore be considered when studying how music can work in this medium.

HEARING AND LISTENING

The way we perceive sound in the real world – and likewise in games – takes two main forms: *hearing*, which is passive and unintentional; and *listening*, which is active and deliberate. A number of game audio theorists have drawn from other bodies of knowledge, some of which have also been used within film sound theory, which discuss this dichotomy of aural perception and have applied it to how players listen to and interpret sound and music in video games. Given that the hypothesis of this research would require a player to listen to game music purposefully in order to gain information, understanding the processes of hearing and listening is therefore arguably important.

Jørgensen (2009: 75–8) points to composer and theorist Denis Smalley (1992), who in turn draws from Pierre Schaeffer, a pioneer in early electroacoustic Western art music, and perceptual psychologist, Ernest Schachtel. Schaeffer was interested in the sometimes unclear relationship between a sound and its source which resulted from the developments in early tape-recording technology of the mid-twentieth century which enabled compositions to be created out of the manipulation of recorded sounds. Schaeffer identifies four modes of listening based on the relationship between listener and source. His first mode of listening is *information gathering*, where the listener attempts to identify the causality of the sound and any information it infers. This was particularly pertinent to Schaeffer's form of *Musique Concrète* which created music from both acoustic and electronic sound recordings that were not necessarily of traditional instruments, meaning that the listener may well have been unfamiliar with the way in which any particular sound was originally created. His second mode is *passive reception*, an unintentional mode where the listener cannot help but hear sounds and which may then lead on to the more active information-gathering mode.

Musical appreciation is Schaeffer's third listening mode. In this mode, the subject listens to the sound itself and its various structures, divorced from any causality. Finally, Schaeffer's fourth listening mode, *musical meaning*, concerns the subject listening to a sound in order to understand a shared musical code. It is important to understand that Schaeffer's four modes of listening are not isolated processes, but that in many ways they overlap and can lead from one to the other. Most often, a subject may begin by unconsciously hearing a sound (mode 2), leading them to focus their attention on understanding its cause and any information it might convey (mode 1), which could then lead to the listener concentrating on specific qualities of the sound (mode 3) and potentially interpreting any musical meaning (mode 4).

Denis Smalley (1992) combines Schaeffer's listening modes with Ernest Schachtel's concept of *autocentric* responses, which are passive, emotional reactions to stimuli, such as music, and *allocentric* responses, which are active and deliberate, thought-based responses. Based on this, Smalley puts forward *relationships* between the listener and a sound (subject and object) which could potentially also be applied to how players listen to game audio. First is the *indicative relationship*, which can be either active or passive, where the subject interprets the sound for its message or information. This is similar to Schaeffer's first mode. Next, the *reflexive relationship* is an autocentric, basic emotional response to the object. This relationship can be active, but most likely will be passive. Finally, the *interactive relationship* is an active, allocentric response where the subject explores the qualities of the object, listening for structures and properties of a sound (Smalley, 1992: 519–20). This relationship covers Schaeffer's third and fourth listening modes.

Other game audio theorists have drawn from composer and film sound theorist, Michel Chion's (1994) three modes of listening – causal, semantic and reduced – as another way of understanding how players listen to and interpret sound in games (see also Friberg and Gärdenfors, 2004). *Causal listening*, which Chion refers to as the most common mode of listening, "consists of listening to a sound in order to gather information about its cause (or source)" (1994: 25); *semantic listening* refers to interpreting a language or code, such as speech; and *reduced listening* concerns itself with the sonic attributes of the sound devoid of other connotations, such as their origin or meaning. However, Stockburger (2003: 4) notes that the reduced listening mode is potentially inappropriate for understanding how players listen to sound in games: "a mode of reduced listening will not be achieved when we are playing an audiovisual game, simply because we are drawn to construct relations between the visual and auditory information we are receiving", and this line of thought could potentially be extended to exclude Schaeffer's third or musical appreciation mode.

Jørgensen (2009: 75–8) argues that Denis Smalley's interpretation of the relationship between listener and sound is more suitable for appropriation to video game audio theory than Chion's listening modes. However, Chion's causal and semantic listening modes arguably elaborate on Schaeffer's *information-gathering* mode and Smalley's *indicative relationship* by dividing them into two distinct parts: the cause of the sound and its semantic message.

THE DIEGESIS

Given that most games are an audiovisual medium, with many being centred on characters and thus focused on narrative, it is understandable that comparisons have been made with film, particularly given that the field of video game studies and likewise game audio are still relatively new areas of academic research and are forced to draw on existing bodies of knowledge. Weinen (2007: 3) wrote that both film and video games "share the common idea of giving an aural background to moving pictures". Likewise, Collins (2005: 1) noted that "Games music shares most of the functions of film music – it helps establish settings, emphasizes movement, and serves narrative functions."

There are, of course, many differences between the two media – most notably the fact that games are primarily interactive systems and not solely linear narratives. As Stockburger (2003: 7) puts it, "In film, the audiovisual contract inscribes static relations between the elements, whereas in games they are dynamic and potentially user driven." However, for the purposes of understanding how sound signals and sources relate to the player of a game, spatial theories of films and film sound can still serve as a good starting point.

The idea of *the diegesis* is one of the main concepts from film studies that has been appropriated for the discussion of game audio. Originating from *The Republic*, where Plato distinguishes between *diegesis* (describing a narrator speaking as himself) and *mimesis* (describing a narrator speaking as a character or not as himself), the term "diegesis" was revived in the 1950s in film studies and has become the accepted terminology to describe the narrative world of a film (see Jørgensen, 2011: 80). This concept can be applied to sound where *diegetic sounds* are understood to be all sounds originating from sources within the depicted fictional world. Likewise, its opposite, *non-diegetic sounds* (sometimes known as *extradiegetic*) are those sounds that do not originate from within the fictional world and therefore would not be heard by the characters within the film. For example, in a car chase scene, the sound of the cars' engines would constitute diegetic sound, while the musical score accompanying this action sequence would be considered non-diegetic.

Likewise, in a video game, a character's crunching of footsteps on the ground would be classed as diegetic sound, while the ominous background music could be considered non-diegetic. However, a complication arises given the nature of the player's influence over the events within the game's world. If, in this example, the ominous music cue foreshadows the presence of a foe, the player, situated *outside* the diegesis, is granted the agency to affect the actions of the character *within* the diegesis, possibly readying a weapon or retreating to safety. If this were a film, however, the viewer would be powerless to change the course of events having heard the non-diegetic music (see Juul, 2001).

The concept of diegesis therefore cannot be applied to video game audio completely unaltered, given the interactive nature of games. As Jørgensen (2011: 81) says, "film sound is limited to informing the audience as to how to interpret what is going on in an inaccessible world while game sound provides information relevant for understanding how to interact with the game system and behave in the virtual environment that is the gameworld." Costikyan (2001) argues from a games studies perspective that, while games can tell stories, they are not principally a story-telling medium. There is an inherent conflict between the way a story works and the way a game works with regards to their structure. "Divergence from a story's path is likely to make for a less satisfying story; restricting a player's freedom of action is likely to make for a less satisfying game … It's not merely that games aren't stories, and vice versa; rather, they are, in a sense, opposites" (Costikyan, 2001: 1–2). Likewise, Frasca (2003) suggests that, rather than narratives, games should be thought of more as *simulations* (see also Frasca, 1999). Frasca argues that the story-telling model is not accurate or useful and suggests games as simulation: "Even if simulations and narrative do share some common elements – character, settings, events – their mechanics are essentially different" (2003: 222).

Many game audio theorists have therefore attempted to adapt the concept of diegesis in order to allow for the interactive nature of video game audio. Ekman (2005) uses the distinction between the source of the sound (its "signal") and what it refers to (its "referent") in order to define four different types of game sound in relation to a spatial understanding of game audio. A sound that has both a diegetic signal and referent she simply terms a "diegetic sound", but a sound that has a non-diegetic signal but a diegetic referent is a "symbolic sound". For example, a symbolic sound could be dynamic music that signals the approach of an enemy. A sound that has a non-diegetic signal and a non-diegetic referent is simply a "non-diegetic sound", but a diegetic signal with a non-diegetic referent she calls a "masking sound". Masking sounds are often messages giving information to the player, but this intention is disguised, or masked, as diegetic sounds: for example, the growl of a monster as the player approaches signals to the player that they are close to this enemy, and is not heard just because monsters growl.

Ekman (2005) identifies that there is a distinction between *outside* and *inside* the virtual world of a video game. Jørgensen (2006, 2009) also identifies this problem and has suggested the term "transdiegetic" to refer to sounds that communicate across the supposed diegetic boundary of a game's virtual world. According to Jørgensen (2006: 51), "Transdiegetic sounds break the conventional division between diegetic and extradiegetic sounds by either having diegetic sources that communicate directly to the player, or by being extradiegetic sounds that game characters virtually can hear." In a similar way to Ekman, for sounds that communicate across the diegetic divide, Jørgensen (2009: 106–7) distinguishes between two types of transdiegetic sound, depending on its source: "external transdiegetic" for a sound that might traditionally be thought of as coming from a non-diegetic source, and "internal transdiegetic" for a sound that might traditionally be seen as coming from a diegetic source.

Grimshaw (2007; see also Grimshaw and Schott, 2007) tackles the concept of diegesis in a way that accounts for the fact that there might be multiple players in a game. His terminology is focused on how sounds within a game relate to each player and, more specifically, how certain sounds have different meaning and relevance, depending on which player is hearing them.

> Thus, we propose the terms *ideodiegetic* and *telediegetic*, the former being those immediate sounds which a player hears and the latter referring to sounds produced by other players but having consequence for the one player (they are telediegetic for that player). Ideodiegetic sounds may be further categorized as *kinediegetic* (sounds triggered by the player's actions) and *exodiegetic* (all other ideodiegetic sounds). (Grimshaw and Schott, 2007: 476)

However, Grimshaw's terminology deals only with sounds that would be traditionally thought of as diegetic and therefore does not address the issue of the player's situation external to the virtual space of the game and their influence within it. Grimshaw's terminology addresses first-person shooter games, and in many ways his model for understanding diegetic sounds essentially treats the player as though they were the avatar within the diegesis.

In "Time for New Terminology?", Jørgensen (2011) moves away from the term "diegetic" altogether, using instead the terms "gameworld" to refer to what might traditionally be thought of as the diegesis and "gamespace" to refer to the other aspects of the game, such as the interface, overlays and menus, that are not easily defined as diegetic or non-diegetic. The "gameworld", therefore, is situated within the "gamespace", and rather than the either/or dichotomy of the diegesis, the difference between gameworld and gamespace allows for the incorporation of the many elements of a game that constitute its interface, which would be difficult to term part of a traditional diegesis and which also include the source of much of a game's audio.

Jørgensen (2011: 92–3) defines five categories based on a spectrum of how incorporated the sound's referent is into the gamespace and gameworld. These categories are: *metaphorical interface sounds* (for example, music representing enemies of the player); *overlay interface sounds* (menu clicks and bleeps); *integrated* *interface sounds* (the sounds of power-ups and buffs); *emphasized interface sounds* (unit responses in an RTS game; see Chapter 3); and *iconic interface sounds* (sounds "natural" to the gameworld such as gunshots and engine sounds). While it could be argued that this model goes too far in considering sounds almost exclusively in relation to visual elements, and that previous models put more focus on the relationship between the sound, its source and its meaning, Jørgensen's categories do highlight how, from a functional standpoint, audio should be considered for its usability function, fully integrated into and key to the information system of a game.

However one considers a game and its sound as divided into separate spaces, what is important to remember is that the way in which players interpret these spaces greatly affects their understanding of information in sound and how it relates to the gameplay as a whole.

ACTIONS AND INTERACTIONS

One of the key aspects that separate games from other audiovisual media is the interactive relationship between the player and the game system. Salen and Zimmerman (2004: ch. 6, p. 2) state that "interactivity simply describes an active relationship between two things", and at a basic level interactivity is constructed of *actions* and *outcomes*. This interaction between the player and the game is often referred to as the *gameplay*, which can be "considered to be [the] core activity of the game which is accessed through the interface" (Juul and Norton, 2009: n.p.). The interface is the tool and bridge with which the game communicates with the player and the player interacts with the game.

A major distinction between games and other forms of media, such as films and novels, which defines the interaction between a player and a game, is the concept of challenge (see Juul and Norton, 2009). Games are goal driven and the difficulty of the goals in a game is spread between its gameplay and its interface. Where the challenges are placed is decided by the game designer. For example, RTS games require strategic and tactical thinking in order to outplay opponents. This is part of the *gameplay challenge*. However, RTS games also require players to select and control many aspects of the game, often performing multiple tasks at a time, which would constitute part of the *interface challenge*. The interface and gameplay are not mutually exclusive aspects of the game and the boundary between them is often blurred (see Juul and Norton, 2009). Juul and Norton note that games are not like most software that is designed to be as user-friendly as possible, and that "games are both efficient and inefficient, both easy and difficult, and that the easiest interface is not necessarily the most entertaining" (2009: n.p.).

Interactivity greatly affects how music and audio in general work in video games. The position of the player as listener changes because of the participatory nature of a game. Wingstedt (2006: 43) suggests that the linear relationship of "creator–performer–receiver", which defines the traditional production and consumption of music, is blurred by the interactive nature of games. Rather than a one-way direction of communication, games afford some of the performance and even, in some cases, creation of their music to the player.

The term "interactive" with regard to game music has been widely used, and arguably misused, having become something of a catch-all term, and even a buzzword for *all* game audio within the industry. There is, however, far more nuance to the concepts behind the non-linear nature of game audio. Collins (2007, 2008b) suggests the alternative term "dynamic" as an umbrella term to refer to all non-linear aspects of game audio: "Dynamic audio, then, is audio which reacts to changes in the gameplay environment *or* in response to a user" (Collins, 2007: 2).

The term "interactive" therefore can be used to describe a more specific characteristic of music and audio in games. Pidkameny (2002) suggests that the term be used to refer to audio that can be affected or caused directly by the *player's* actions (see also Collins, 2007, 2008b: 4). Some audio, however, can change with the events of the game but not as a direct response to the player's input and actions, and so should not be thought of as "interactive". This type of audio has instead been referred to as "adaptive". "Adaptive audio ... is sound that reacts to the game states, responding to various in-game parameters such as time-ins, time-outs, player health, enemy health, and so on" (Collins, 2008b: 4; see also Whitmore, 2003; Kaae, 2008). Adaptive audio is not something new to games. Even *Space Invaders* in 1978 incorporated what could arguably be thought of as adaptive audio. The background pulse speeds up as the player progresses through the game, increasing the tension of the experience (see Pidkameny, 2002; Collins, 2005: 2).

It is therefore the cause or trigger of a sound event that determines how the type of dynamism should be described. Jørgensen (2008a) distinguishes between a sound's "source" and a sound's "generator". The *source* is the object from which the sound emanates, while the *generator* is the thing that causes the sound to be heard. In the case of "interactive" audio, the generator has to be the player, while in "adaptive" audio, the generator is the game itself. While player input may often seem to cause audio events to play, there should be a distinction between how direct or indirect the link is between this input and the sound event. This can often be determined by the intention of the player's input. For example, when a player presses the "B" button in *Super Mario World* (Nintendo, 1990), Mario will jump and a cartoonish "bleep" will be heard. This sound effect could arguably be called "interactive" by many theorists' definitions, but given that the intention of the player by pressing the button was for Mario to jump, and not to play a sound, this audio event should potentially be thought of as "reactive" as it is in response to a player action that was not directly intended to interact with the game's audio.

There are many types of game where the intention of the player's input *is* to interact with the game's audio directly. Games such as *Rez* (United Game Artists, 2001), *Guitar Hero* (Harmonix, 2005) and *Electroplankton* (Indieszero, 2006) are music-based games where the focus of gameplay is towards players directly affecting the playback of the soundtrack (see Pichlmair and Kayali, 2007). This is a type of truly interactive audio in its most literal sense.

A final type of dynamic audio that can be in itself adaptive or interactive is *generative* audio. Often music is considered generative when some aspect of its composition is determined by the game at runtime. Also sometimes referred to as "variability" in game music (see Kaae, 2008: 83–4), the game's code may use some rules or random chance in order to make decisions about the music's playback. This is also a useful technique to employ in order to create as much variation as possible within a game's soundtrack.

Most dynamism within game audio is due to the inherent non-linear nature of video games, and most of the time players do not think twice about audio following the dynamic actions of the game. Sound effects, such as gunshots and car engines, endeavour to mimic the way in which sounds work in the real world, but many aspects of video games, music in particular, do not have a real-world analogue, but tying these features to the game's actions can be highly beneficial to the player's experience. Whitmore (2003: n.p.) argues that non-dynamic music is analogous to pre-rendered visuals and dynamic music is therefore the equivalent of real-time graphics. Few people today would accept games constructed purely from pre-rendered video sequences for their visuals, but this is essentially the attitude of many towards the treatment of music in games.

STRUCTURES

The previous sections have looked at games as interactive systems, often represented as fictional worlds, but most games are not completely open sandboxes: games are structured systems, and players' actions are restricted by the game's rules and these limit the possible courses of events within a game. "Game structure has to do with the means by which a game shapes player behavior" (Costikyan, 2002: 20). Far from being a negative aspect, this is what defines the way in which the game is played: "The enjoyment of games hinge[s] on their rules, not on their representational level" (Juul, 2004: n.p.). Unlike physical games, where rules will often be written out as instructions, the rules of a video game are part of its software and players will generally learn these rules by playing the game. It is perfectly possible, in many cases, for players to play and enjoy a game without entirely understanding its rules, "instead gaining a 'gut,' intuitive understanding of their operation" (Costikyan, 2002: 19).

Music in video games is likewise structured, and designers can use the rules and limits of what players can do within a game in order to structure a game's music. The rules create the dynamic flow of the game and therefore the pacing is set by the player's actions. Time therefore plays a pivotal role in the way in which music is structured within a game. Kaae (2008) notes that the perception of time can be interpreted differently depending on whether a person is experiencing it in the present or whether it is being considered in the past or future.

As it is being experienced in the present, from the listener's perspective, music always appears to be "linear" (Kaae, 2008: 77–8): one section is always followed by another, section A might be followed by section B and so on. But from any other perspective, music can potentially be structured in a "non-linear" fashion, especially in a dynamic medium like a video game. Section A could be followed by either section B1 or B2, depending on the player's actions. Whether or not the music is thought of as linear depends on which viewpoint the music is being considered from. Kaae (2008: 78) uses the term "multi-linear" to describe the way in which music can be structured non-linearly but experienced in a linear fashion.

In linear audiovisual media, the composer always knows what is going to happen at any particular point in time and so transitions can be predetermined, just as the rest of the music is predetermined. But depending on the degree of agency a player is given within a game, one action might lead to two drastically different outcomes, requiring one music cue potentially to be able to transition to two or more other music cues that may be very different from one another. There are a number of ways, therefore, that music can transition from one cue to another. The most basic of these methods is *direct cutting* where, at the moment of transition, the first cue is stopped and the second is started. This, however, can create a jarring experience for the listener and is generally not used in modern video games. A more subtle version of this is the use of *cross-fades*, which has the effect of smoothing over the point at which one cue ends and the other begins. Likewise, musical *stingers* (very short stabs of music) can be used to mask the point of transition between two cues (see Collins, 2007: 5). Although very time-consuming, composers can potentially create a short transitional cue for each possible transition in the music (see van Geelen, 2008: 96). Collins (2008b: 160–3) also refers to this as a "transition matrix", although she points out that the amount of time needed for planning and execution can be very prohibitive.

Beyond simple cue-to-cue transitions, there are two overarching categories of dynamic musical change in the structure of game audio: horizontal and vertical. These terms refer to how music is visually expressed on sheet music (see Kaae, 2008: 87; Phillips, 2014: 187–8, 193). Horizontal changes are those aspects of music that occur across the page: for example, tempo, time signatures, sections, bars and rhythm. Vertical changes, on the other hand, occur up and down the page: for example, instrumentation, texture, harmony and dynamics.

Horizontal structure, sometimes referred to as branching or node-based music, uses many of the transitional techniques discussed above and is a popular way of controlling dynamic music in games. It features distinct chunks of music that, through the logic of the game's code, are connected to one another in a similar way to the branches of a tree so that one musical cue has the ability to transition to a number of others depending on the circumstances of the game. Collins (2007: 5) likens this musical structure to a complex urban metro train system where "A player may at any time jump train cars, stop at stations, get on new trains, or change the direction of the car they are in." Kaae (2008: 77) uses the term "hyperstructure" to refer to this musical form where the pieces of music are "nodes" and the triggers that cause transitions to occur are called "links". The goal of this type of music is therefore to match the events of the game with the links in the hyperstructure so that each node or piece of music fits with the actions of the game. Transitioning between nodes is an important aspect of this style of music. As Collins (2007: 5) states, "Moving smoothly from one music cue to another assists in the continuity of a game, and the illusions of gameplay, since a disjointed score generally leads to a disjointed playing experience, and the game may lose some of its immersive quality."

Time plays a significant role in how a node-based soundtrack can transition. Depending on the points at which the music can progress, there may well be a certain amount of delay between when a gameplay event occurs that would cause a change and the music reaching a point where it is able to transition. For example, if the music can only progress at the end of a node, the length of the cue will determine how quickly the soundtrack can respond to the gameplay. Many node-based soundtracks are able to count bars and beats and interrupt themselves mid-way through a cue and therefore follow events with much less delay, but there will always be some delay due to the nature of musical timing (see Stevens and Raybould, 2011: 193–212). "Nevertheless, this approach is increasingly common, as composers write literally hundreds of cue fragments for a game, to reduce transition time and to enhance flexibility in their music" (Collins, 2007: 5).

Vertical structure, sometimes referred to as *layering*, *vertical layering* or *variable mix*, is a type of dynamic music control that focuses on the manipulation of individual instrument parts or layers in order to follow actions within the game (see Collins, 2007: 5–6, 2008b: 152–5; Stevens and Raybould, 2011: 212–27; Phillips, 2014: 193–201). While horizontal structure creates changes in the music by moving from one cue to another, vertical structure creates its change through the addition and subtraction of layers of music within a single cue with the aim of creating distinctly different musical states through combinations of the layers. While there are plenty of examples of vertical layering within games, this type of dynamic music, which is the main focus of this research, has not received as much attention as horizontal structures.

The use of vertical layering provides many advantages to the designer and composer. Although almost any parameter of music can be affected by vertical layering to some degree, instrumentation, texture and dynamics in particular can be easily manipulated by this type of musical control. This is because these parameters of music are quite *time independent*. For example, while a bar line happens at a very specific moment in time that cannot be changed without completely changing the structure of the music, a change in volume, for instance, can happen at any time without affecting the overall flow of the music (see Kaae, 2008).

Vertical control can be as simple or complex as the composer desires: the music could consist of one main part and an additional percussion part to add drama, but could equally comprise dozens of instrument layers, allowing for a vast array of different combinations that could in theory match greatly different gameplay situations. It should be understood that horizontal and vertical structures do not necessarily have to be mutually exclusive techniques for musical control. In a sense, vertical layering could be considered dynamism within an individual music cue, while horizontal structure could be thought of as dynamism on an overall macro-level.

There is huge potential for creating vastly different types of dynamic musical structure in a game's soundtrack using horizontal and vertical techniques and, as Collins (2008b: 164) notes, "[c]ertain genres lend themselves well to certain dynamic styles of music" and that "[t]hese approaches require new ways of thinking about music."

INFORMATION SYSTEMS

Salen and Zimmerman (2004: ch.17, p. 8) define information in games as "knowledge or content that is manipulated, acquired, hidden, and revealed during play."

Information within a video game can be known to all players, only certain players, known only to the game system or randomly generated on the fly. Salen and Zimmerman (2004: ch. 17) refer to this as *perfect* and *imperfect* information within games, the former being where all information is known to every player (as in chess and backgammon) and the latter being where certain information is hidden from some or all players (as in poker and *Cluedo*). Ultimately, the value of information in games is created by its context within the game and in comparison with other pieces of information.

While these principles are applicable to all types of games, video games are able to make particular use of complex systems of information due to their ability to compute large amounts of data without the player ever knowing what is going on behind the scenes, and often the discovery and use of information can play a large part in the gameplay and be a contributing factor to the appeal of the game. "Many digital games rely on vast sets of information rewards for player interaction. Huge worlds to explore, complex economies of items, and hidden fighting moves are the 'stuff' with which digital game designers fill their systems" (Salen and Zimmerman, 2004: ch. 17, p. 8). Salen and Zimmerman (2004) argue that, in order to create a meaningful experience for players, information must be discernible and integrated within the game. To this end, given that all music can potentially communicate to the listener, it follows that music, particularly when it is dynamic, should be integrated into the information system of a game, otherwise the player is being provided with information of low importance to their gameplay experience.

Dynamic soundtracks can be controlled using information or gameplay data fed directly from processes within the game's code. Kaae (2008: 88) identifies two categories of gameplay information which he refers to as "discrete messages" and "continuous messages". *Discrete messages* are individual events sent from the game engine that represent a single moment in time. These can be used to trigger transitions in music and are particularly useful for horizontal structures, but can also be used to set what might be thought of as a *mix state* of a vertical soundtrack. For example, a player achieving an objective, entering a new area or levelling up their character could each create a discrete message to be sent to the music system. *Continuous messages* represent a constant stream of information coming from the game engine. Sometimes referred to as *runtime parameters*, continuous messages are particularly useful for controlling vertical layering music cues given that this type of information can be tied to the volume of certain musical layers "and thereby follow game actions at a very detailed level because of their indefinite variability" (Kaae, 2008: 90). For example, the numerical value of a player character's health could be linked to the volume of a percussion layer where the lower the health gets, the louder the percussion is heard in the mix, increasing the sense of danger.

Auditory display studies is a field that looks at the use of sound as a means of communicating information, usually within the context of a user interface and computer interaction (see Kramer, 1994). An auditory display is therefore the sonic equivalent of a visual display like a computer monitor. Knowledge from this field has significant relevance to understanding how music can function as part of a video game's information system. Within the field of auditory display, the concept of "sonification", generally defined as the use of non-speech audio to represent information (see Kramer et al., n.d.), can be considered a more specific process of turning data into sounds "that exploit the auditory perceptual abilities of human beings such that the data relationships are comprehensible" (Walker and Nees, 2011: 9). As Kramer et al. (n.d.) note, "environments in which large numbers of changing variables and/or temporally complex information must be monitored simultaneously are well suited for auditory displays." Kramer could just as easily be describing a video game here, and therefore it could be argued that the fields of sonification and auditory display offer potentially valuable ways of understanding how music in games can work as an information system. Grimshaw (2007: 115) likens the whole game engine to "a sonification system in that it translates non-audio data (the game status or player actions, for example) into sound through the agency of the computer's or console's audio hardware thereby providing sonically interpretable data to the player."

There are many benefits to the use of auditory displays. They can allow the user simultaneously to pay attention to visual elements; they can serve as alerts and quickly draw the user's attention; they can help orientate the user; more than one audio source can be attended to at one time, allowing for comparative listening; and changes in sounds over time can be heard at a high level of acuity (see Kramer, 1994: table 1; see also Heeter and Gomes, 1992).

While the idea of turning data or information into sound may seem like an unusual concept, Kramer (1994) argues that in many respects it is not very different from the way in which a musician creates music: "Assembling, finding, and manipulating sonic materials: This could describe auditory display design. It could just as well characterize music composition and performance" (Kramer, 1994: s11.2). In this way, the musician's intentions are the data and the instrument is the object that sonifies this data. Humans have devised many ways of organising sound and dividing it up into structures. The Western musical tradition, for example, has developed diatonic scales, common metres and a wide assortment of standardised musical instruments. Musicians also draw inspiration, and therefore in a sense data, from various aspects of the world. Art, architecture, nature and even randomly created data from the roll of a dice or the draw of a card have been combined with the aforementioned structures in order to create music. There is therefore much overlap between music and auditory display and many skills are shared between the two fields.

When representing information through sound, it is important to consider what sonic parameters are used. Certain aspects of sound may be better than others at representing particular types of information (see Walker and Nees, 2011: 24). According to Kramer (1994), there are two broad categories of sonic data representation which occupy two ends of a spectrum: *analogic* and *symbolic*. At one end, *analogic* representation "is one in which there is an immediate and intrinsic correspondence between the sort of structure being represented and the representation medium" (Kramer, 1994: 21). For example, the clicks of a Geiger counter represent the presence of radiation: the higher the number of clicks, the higher the amount of radiation. At the other end of the spectrum is *symbolic* representation where there is no link between the sound itself and the information it is representing. For example, a fire alarm has no direct connection to fire and is an arbitrary sound we have learned to associate with danger in a certain situation.

There are three main functions of auditory signals that are referred to within the field of auditory display which are also in many ways applicable to sound in video games: proactive, reactive and monitoring functions. Sounds that have a *proactive* function often convey information to the user that needs to be addressed. Alarms, alerts and warnings are all examples of auditory signals with a proactive function. They are initiated by the system and often require a response from the user (see Jørgensen, 2006: 49; 2009: 62; Walker and Nees, 2011: 13). Sounds that have a *reactive* function serve as responses to user actions. Button presses and other similar interface sounds inform the player that their actions have been recognised by the system (see Jørgensen, 2006: 49; 2009: 62). Finally, sounds with a *monitoring* function can allow changes within the information they are representing to be heard by the user as changes within the sound; for example, the pulse sound of a heart rate monitor speeds up and slows down in relation to the heart rate of the patient. It should also be noted that this monitoring function can take a more interactive form and rather than simply displaying data for the user, it can allow them actually to explore this data (see Walker and Nees, 2011: 13–14).

Already, aspects of auditory display studies have been appropriated for use within the field of game audio. One such aspect is the relationship between an auditory signal and the information it represents. The terms *auditory icon* and *earcon* are used within auditory display studies to refer to two variations of this relationship. Jørgensen (2006: 49) describes auditory icons as "characteristic sounds based on a principle of similarity or direct physical correspondence and which can be recognized as sounds connected to corresponding real world events". Essentially, auditory icons are the sounds that the source might be expected to make if it were a real world object and often utilise actual sound recordings of these objects, though they could equally be caricature-like versions. For example, most game sounds that might traditionally be termed "sound effects", such as gunshots and footsteps, could be thought of as auditory icons (see also Friberg and Gärdenfors, 2004; Grimshaw, 2007: 110; Jørgensen, 2009: 87–8).

At the other end of the spectrum are *earcons*. Jørgensen (2006: 49) defines earcons as "symbolic and arbitrary sounds such as artificial noises and music which may be seen as abstract in the sense that they cannot immediately be recognized".

Earcons, therefore, have an arbitrary or no direct relationship to the information they represent. For example, music that signals the approach of an enemy or the buzz of a rejection notification does not directly mean anything about its cause; instead, the relation to its referent and thus its meaning is learned by the player (see also Friberg and Gärdenfors, 2004; Jørgensen, 2009: 87–8).

Jørgensen (2009: 90–2) adds a middle category of *non-arbitrary* relationship to refer to a sound that is related to, but not a literal representation of, the sound that would be created by its source, but that is not an arbitrary reference like an earcon. She gives the example of the sound that is heard when a player selects one of their buildings within the RTS game *Warcraft III* (Blizzard, 2002). When selecting a barracks, the player will hear the sound of marching soldiers. Given that this sound is only heard briefly after the building is selected, it is not literally meant to be a sound natural to the environment, but at the same time it is not a sound arbitrarily assigned to the action of selecting the building.

The ability of listeners to learn the meaning of sounds is therefore particularly crucial to the use of earcons. Hermann et al. (2011: 3) state that a "fascinating feature [of using sound to convey information] is the ability to learn and to improve discrimination of auditory stimuli ... Expertise in a particular domain or context can dramatically affect how meaning is constructed from sound." This is particularly important for earconic sounds as, while in the case of music the designer and user may share the knowledge of many sociocultural conventions which have been learned over time, generally speaking the relationship between the signal and the referent, in the case of earconic sounds, is often arbitrary (see Grimshaw, 2007: 114). As Friberg and Gärdenfors (2004: n.p.) put it, "All interface design is about establishing agreements between the designer and the user. This gets easier over time, as conventions develop."

However, Oswald (2012) suggests a revision of auditory icon and earcon theory based on semiotics. He argues that defining auditory icons as being purely based on similarity with real-world sounds, and earcons as completely arbitrary sounds, is too simplistic a division. He argues that the two are not completely separate in their representation of meaning through sound. While it has often been assumed that musical earcons convey their information by being arbitrary signals whose connection to their information is entirely learned, in isolation and/or through layers of previously learned cultural conventions, there are a number of ways in which sound, and particularly music, can potentially be understood by listeners without prior exposure to the signal. As Oswald argues:

> there are also aspects in music that are directly understood, independent of musical training and across cultural differences. These so-called musical universals are based on biological and physiological structures, or rooted in human perception. The sense of tempo correlates perfectly with both heartbeat and walking; 120 beats per minute are considered a fast tempo in music, a fast heartbeat rate, and also a fast walking pace. Universal music related patterns are also found across different spoken languages. An excited speaker will speak louder and faster, in a higher pitch, using greater intervals – features that are also used to describe excitement in musical theory. (Oswald, 2012: 40)

The way in which sound and music can contain or be ascribed meaning and information has also been discussed by Walter Murch (1998). Murch suggests that a spectrum from *encoded* meaning to *embodied* meaning can be used to describe this concept. The most obvious example of encoded sound is speech where the meaning of the sound takes the form of a code or learned system: the actual sounds have an arbitrary relationship to their meaning. At the other end of the spectrum, music exemplifies embodied sound, the meaning of which is contained within the sound itself and not conveyed by any learned code. Murch (1998) notes, however, that while music is often referred to as a "universal language" because of its potentially embodied meaning, music is very often influenced by learned cultural conventions which push it towards the encoded end of the spectrum. Likewise, unless spoken monotonously or completely computer-generated, speech usually contains musical qualities which push the sound towards the embodied end of the spectrum. While music and speech are the two extremes, any sound can be described on this spectrum of encoded and embodied meaning based on its semiotic and musical qualities.

USABILITY FUNCTION

Information can be conveyed through sound and music within the context of a game in much the same way as theorists have discussed and designers have used sound within the field of auditory display. While the function of audio that creates a sense of presence and immersion for the player of a video game has been discussed at length within the field of game audio, the research presented here focuses on a second key function of game audio: *usability* – how audio aids a player's interactions with a video game system through the conveyance of information.

Sound in games has the ability to alert players to events and actions that may be about to happen or currently happening out of the player's line of sight. Collins (2008b: 129–30) refers to this as the *preparatory function* of game audio. "A crucial semiotic role of sound in games is the preparatory functions that it serves, for instance, to alert the player to an upcoming event. Anticipating action is a critical part of being successful in many games, particularly adventure and action games" (Collins, 2007: 8). Jørgensen (2008b: 167) adds that this is a particularly useful function of game audio when the visual system is busy or otherwise unavailable.

Game audio also has an *identifying function* which helps the game to be understood by the player (Collins, 2008b: 130-1). According to Collins (2007: 8), "symbols and *leitmotifs* are often used to assist the player in identifying other characters, moods, environments and objects, to help the game become more comprehensible and to decrease the learning curve for new players." Jørgensen (2006: 51) adds that sound can even help ascribe *value* or worth to its source. Musical themes can also help the player understand their situation within a game's various environments and, as a subset of the identifying function, audio can also serve to aid *navigation* within a game "because users associate it with their position in the interactive flow" (Hoffert, 2007: 56). "Even when music isn't consciously used for navigational design, it functions that way ... [players] soon associate the different music cues with characters, scenarios, and levels" (Hoffert, 2007: 58; see also Grimshaw, 2007: 101; Grimshaw and Schott, 2007: 477). *Emotional meaning* is game audio's final main usability function. The emotional aspects of sound, and particularly music, are often discussed in terms of creating mood and atmosphere in a game. However, as Collins (2007: 10) explains, sound has the ability to communicate information about emotion without itself affecting the player's emotions. A player can understand a circumstance that depicts anger without themselves feeling angry.

Unlike traditional linear media where sound, and particularly music, are frequently considered a background element, often passively and subconsciously absorbed by the audience, listening to and understanding game audio is a highly important part of successfully playing video games, given that games are an interactive medium and that game audio can contain pertinent information (see Collins, 2007). Multi-player games, in particular, highlight the need for players to master all aspects of a game, including any information conveyed by its audio. If one player is using the game's sounds as a source of information, then all players need to in order to stay competitive (see Ekman, 2005). As Grimshaw (2007: 94–5) argues, learning the meaning of sound is a vital part of learning a new game. It "leads to the acquisition of experience and means that, in later playings of the game, this experience can be brought to bear to interpret that sound with immediacy."

3

Audio in Real-time Strategy Games

> *[V]ideo game systems and games themselves are the starting points of theories. They have influenced and will continue to influence the methods of looking at video games.* (Perron and Wolf, 2009: 2)

This research looks at the use of music as a means of conveying information in video games and has been framed within the context of real-time strategy (RTS) games because of their analytical and information-rich characteristics. While Chapter 2 explored theories of how audio works structurally and functionally in games, this section will use this understanding to look specifically at the audio of actual modern RTS games, using *StarCraft II* (Blizzard, 2010), *Dota 2* (Valve, 2013) and *Strife* (S2 Games, in development) as examples of the genre.

RTS games generally take the form of some sort of conflict, often involving real-world, fantasy or sci-fi archetypal armies. For example, *StarCraft II* features three different factions (Terran, Zerg and Protoss), and the sound and music are themed towards their aesthetics. RTS games are played from a top-down perspective with the player taking a god-like view over the action. Players will often be required to build bases and control their troops, referred to as *units*, in order to defeat their opponents' forces. The graphical interfaces of RTS games generally feature a considerable amount of gameplay relevant information and ways to control and interact with the game (see Figure 3.1).

Figure 3.1 Screenshot of *StarCraft II*, showing top-down perspective, units and user interface.

Audio can vary greatly from game to game, but there are many common conventions across the genre. This discussion looks at the main gameplay features of RTS games and how audio works in conjunction with them. It is with an understanding and mastery of these gameplay features, which can include the game's audio, that a player defeats his or her opponent.

GAMEPLAY

There is a concept known as *macro-management* within RTS games which refers to the ability of the player to attend to the large-scale goals of the game. The main aspect of macro-management is the ability of players to collect and spend *resources* which enable them to carry out their strategy and ultimately defeat their opponents. What is classed as a resource will vary from game to game, but it is generally either something physical, such as timber, minerals and gold, or something abstract, such as experience points. In *StarCraft II*, autonomous workers collect resources (see Figure 3.2); in *Dota 2* and *Strife*, players actively collect resources themselves through killing enemies.

Figure 3.2 *StarCraft II* worker and mineral resource.

In terms of audio, the main way in which the player is informed and receives feedback about these macro-level game objectives is through vocal announcements. These are often disembodied voice-overs that are likely to be themed to the game's setting or situation of play, but do not come from a visible character, and therefore not clearly from the gameworld. For example, in *StarCraft II*, the line "Mineral field depleted" indicates to the player that they need to start mining elsewhere. These proactive auditory signals are particularly useful as a player will not be continually overseeing their mining workers.

Non-speech sound cues are also often used to inform the player about resource management. These can take the form of auditory icons and earcons and, as macro-related events are of high importance to the player, they are almost always heard in a non-spatialised manner; that is, they are not positioned in the three-dimensional space of the gameworld, but heard equally over the stereo speakers, and also do not have the environment-specific acoustic properties applied to them. For example, if a player were to hear the sound of coins clinking into a purse with no specific origin in *Dota 2*, they would understand this non-arbitrary signal to mean that they have received a bonus amount of gold.

In RTS games, the opposite concept to macro-management is micro-management. This refers to the ability of a player to understand and control the small details of a game and, in particular, relates to the idea that in order to perform well in a game, a player must be able to execute multiple, small-scale actions at any given time. Their ability to do this will give them a significant advantage over their opponent.

Micro-management is particularly important with regard to controlling units. These troops and vehicles can vary greatly from game to game, but generally can be controlled individually or in groups, using the mouse buttons. They will be likely to have the ability to perform actions, such as moving and attacking, as well as potentially having special abilities, such as using rare weapons or magic spells (see Figure 3.3). Audio plays an important role when it comes to the control of units in terms of informing the player about their actions.

Figure 3.3 A hero from *Strife* using a special ability.

Unit response sounds are the *reactive* sound cues that play as a result of the player interacting with a unit. When a unit is selected or issued with an order, the player will hear a sound cue that confirms the action. For example, this might be the vocal response "On the move" from a character in *Strife* or the alien chittering of a Zerg drone in *StarCraft II*. Here, both examples are vocalisations, but only the former contains semantic meaning. The fact that they both serve the same purpose shows that the signal itself contains the information and not its semantic content. Unit responses are an important feature of micro-management as it means that the player is given instant feedback for their actions. Units in an RTS game are often grouped closely together; therefore, if the player clicked on the wrong unit, they would quickly realise their mistake given the unit response sound.

Despite the fact that unit response sounds appear to originate from units within the gameworld, they are mostly non-spatialised and are just as audible when the camera is nowhere near the triggering unit, although some subtle, non-realistic panning may be applied. In many RTS games, a selected unit will be visually represented in the heads-up display (HUD) by an animated portrait (see Figure 3.4). For vocal unit response sounds, this portrait will often be lip-synched with the audio, creating an interesting relationship between the gameworld and the HUD with regards to understanding the different virtual spaces within a game. Likewise, when multiple units are selected at once, only one unit response sound will be heard. This is especially noticeable when multiple types of units have been selected: generally, the game engine will prioritise the most important unit within the group and use that response sound. Finally, each player can only hear their own units' response sounds and not those of their opponents.

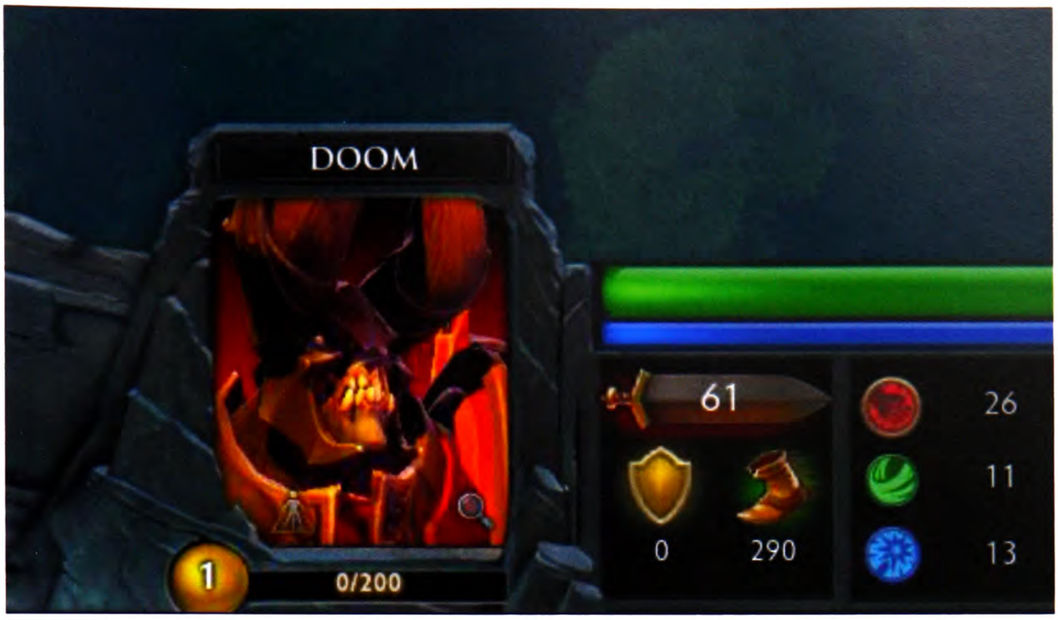


Figure 3.4 A hero portrait from *Dota 2*.

The above points all illustrate the concept that unit response sounds are a usability function intended to aid the controlling player and are not primarily there to be the realistic sounds of the gameworld. Furthermore, *control groups* can be used to select units via the keyboard even when the player is looking elsewhere. The unit response sound in this case is even more important in a usability sense as the player only has aural feedback confirming that they have selected the intended unit. This highlights the identifying functions of unit response sounds.

All of the sound cues that occur directly as a result of the virtual *actions* of a unit – for example, gunfire, magic spells, explosions or a spaceship's engines – could potentially be termed *unit action sounds*. While these sounds might usually be referred to just as sound effects, it is useful to distinguish them from other sounds in a game (such as response sounds and interface sounds) as they are generally audible to all players and are much more spatialised within the gameworld; that is to say, moving the camera will affect the direction from which the sound emanates, and the distance from the camera will affect the attenuation of the sound.

Although it may seem as though unit action sounds are used purely to create a more realistic audio accompaniment to the visual, there are many instances in RTS games where we see that this is not the case. Often, realism is sacrificed in order to increase the usability of the game for the player. One example of this can be found in *StarCraft II*, where vehicle engine sounds will only last for the first few seconds of the vehicle starting to move. Even if the unit continues to move after this time, there is no more need for the sound from a usability standpoint. In a game where there might be hundreds of units moving around at any one time, the player would be completely overloaded with sound if they all continuously made movement sounds.

Dota 2 is another game where this realism is often sacrificed for usability. In *Dota 2*, when units attack each other, the player hears unit action sounds much like in any RTS game. However, when the most important unit the player controls, their *hero*, is attacked, they will hear much louder and more exaggerated impact sounds. The fact that the hero is in danger is deemed more important than other aspects of the soundscape and so is emphasised. This again illustrates the idea that the game's aural realism often comes second to the use of audio as an important part of the interface of the game.

Most RTS games feature some form of buildings or building-like constructs that make up the player's or enemy's base. In terms of audio, structures mostly follow similar rules to units. An additional aspect of structure-related audio that should be noted are *construction complete* sounds. These are the sound cues heard whenever a structure has finished a task given to it by the player. These function similarly to unit response sounds. However, unlike unit response sounds, which supply the player with immediate feedback on their actions, construction complete sounds are proactive signals designed to inform the player about a process they initiated earlier in the game. Therefore, a construction complete sound is often very useful for the player as their concentration may be elsewhere when the task is complete. For example, in *StarCraft II*, a player may have started to build a powerful siege tank, and in the meantime he notices his units are under attack outside his base. While he is attending to that skirmish, he hears the rumble of tank tracks, a non-arbitrary signal which informs him that his tank is complete. He is then able to use the tank to help win the skirmish. Had he not received this audio cue, the player might have been too preoccupied and not remembered the tank or noticed its completion. Certain construction complete sounds may even have a preparatory function where the sound cue begins to play a certain number of seconds before the unit or structure has been completed. Players familiar with the cues will therefore be able to anticipate the completion of the process and prepare to act immediately.

We can see from this section that while audio relating to micro-management uses a mixture of non-spatialised and spatialised sound cues, they are mostly attached or related to an object within the gameworld, although the often non-realistic spatial treatment of these sound cues, and their various positions on the iconic/earconic spectrum, highlight their functional qualities as part of the usability system of the game.

RTS games are usually played across defined levels consisting of terrain that can include hills, cliffs, rivers and trees. Levels may feature general stereo ambiences or more specific sound emitters tied to the scenery that makes up the levels, but both are designed to create a realistic or caricatured sound environment to match that of the visual. Although much of the level environments will be there for aesthetic purposes, RTS games will often simulate aspects of real warfare. One such mechanic, referred to as *fog of war* in RTS games, simulates the idea of only having vision where you have troops or other forms of reconnaissance (see Figure 3.5). This has particular relevance to the audio because, as well as not being able to *see* past the fog of war, the players also cannot *hear* past it. However, in some cases, the game designer may deliberately allow certain sounds to be audible, even when in the fog of war, to give the player extra information through the game's audio, essentially giving these sound cues preparatory and identifying functions.

Figure 3.5 Fog of war in *Dota 2*.

MUSICAL SOUNDTRACK

Much of the emotional atmosphere of an RTS game is created by the musical soundtrack, usually through the use of Western compositional conventions. The music will often be designed to fit with or represent a particular narrative element, such as a faction, an environment or a time period. Compared to many other genres of video game, the music of RTS games will often be less dynamic in its implementation, meaning that the music may not react or relate to real-time events within the game. The less reactive the music is to the gameplay, the less information it transmits that is relevant to the playing of the game. Because of this, and because it could potentially mask more important sounds, many players choose to turn off the music of RTS games. A player has to process a lot of information while playing an RTS game. If either the visual or audio channel is giving them information that will not help them to win the game, then this information will be deemed unnecessary.

StarCraft II

StarCraft II can be played in a number of ways (campaign story mode, multi-player or custom games) and how the music works depends on the mode. This analysis focuses on the multi-player mode as its limits are more definable and comparable with the other games discussed here. The soundtrack of *StarCraft II* is mostly made up of individual linear pieces of music that are organised in the form of playlists. Each of the three factions has music that is themed to their aesthetic, setting the mood for the particular faction currently being played. The human Terrans follow an American space-western aesthetic, and therefore their music is inspired by country and blues and features orchestral elements augmented with guitars and other folk instruments. The Zerg are a species of insectoid, hive-minded aliens, and their soundtrack is composed of strange organic and synthesised sounds. Finally, the Protoss are a highly spiritual, telepathic, humanoid alien race and their music is orchestral but also choral to reflect their religious and ethereal nature.

At the start of a game, a track from the particular faction's playlist is randomly selected to play. Between tracks there will be pauses of certain lengths in order to create contrast within the soundtrack and so the music is not continually playing. However, there is a tick box in the options menu that turns on continuous music. This setting highlights the fact that the developers recognise that players' musical preferences may vary, thus allowing for no music, some music and continuous music.

Each game of *StarCraft II* ends with a musical victory or defeat *stinger* which serves as musical closure to the match and, if the player has won, acts as an aural reward.

The music of *StarCraft II* is therefore mostly non-dynamic given that events within the game are in no way reflected by the music itself. The music serves only to reinforce the theme and nature of the faction currently being played. The only small aspect of dynamism within the music actually serves to emphasise how other aspects of *StarCraft II*'s audio are prioritised for their informative nature: the music is temporarily reduced in volume when an alert cue plays.

Dota 2

In contrast to *StarCraft*'s playlist structure, the music of *Dota 2* is a single, but more complex dynamic system that is themed to the gameworld rather than to the specific factions or characters within it. *Dota 2*'s soundtrack is particularly pertinent to this research as parts of it use a layered approach to structuring its music cues. Explained in a personal communication from Roland Shaw, sound designer at Valve, the music system of *Dota 2* is made up of a number of interrelated music cues. Games always start with an introductory "pre-battle" cue and finish with either a win or lose cue. In between this, the music system chooses one of two main types of cue depending on the amount of action currently taking place. As Shaw explained, there is a "battle value variable" that increases as players use their special abilities and deal damage. When the battle value is low, one of the many "explore" cues plays. As long as this value remains low, explore cues will continue to play with random sections of silence between them.

These cues are made up of three individual layers, the first of which always plays when the cue is called, while the other two are introduced and raised in volume as the battle value increases. When the battle value reaches a certain threshold, the system ends the explore cue and starts the battle cue. This music emphasises the intensity of moments of high action within the game and the system constantly checks whether the battle value variable is still above its threshold, in which case it will continue to play the battle cue. So that the system does not pre-emptively respond to momentary lapses in action, the system waits a small amount of time after the battle value has dropped below the threshold, and if it is still below after this time period, a final "battle over" cue is played. It should be noted that the battle value, and therefore the state of the music system, are specific to each player relative to the activity around them.

The music system has a separate section of music that plays when the player's hero has been killed and is waiting to re-spawn (revive and re-enter the game). This cue begins with a stinger that signifies that the hero has been killed and ends with a longer stinger that proactively anticipates the revival of the hero. Finally, the music system also reacts when a player attacks the boss-like map objective, Roshan. Whichever team defeats Roshan gains a significant advantage over the opposing team, and therefore, to reflect the importance of this, the music system switches to a special Roshan battle cue and finishes with a stinger when Roshan has been slain.

When asked about his thoughts on the dynamic music in *Dota 2* reflecting in-game action, Shaw stated that he felt that the current system mostly fulfils their goals for the music, although he noted that a potential enhancement to the system would be to enable the music to reflect the scale progression that happens over the course of a match.

> One future consideration we have is altering the macro progression of the music over the course of a game. An early skirmish can currently trigger the same battle music as an epic base battle with huge plays and big ultimate abilities. The intensity of the SFX will scale into that just by being attached to abilities, so the mix may get a bit chaotic if we ramp the music up farther, to the detriment of the gameplay. (Shaw, personal communication, 2013)

Dota 2 is a re-make of the *Warcraft III* (Blizzard, 2002) popular mod, *Defense of the Ancients*, and as such comes with many player expectations about how the sequel should work, and, more specifically, how it should be as close as possible to the original. Therefore, all gameplay relevant information is conveyed almost exclusively via non-musical audio, although there are a number of musical stingers that act in a similar way; for example, when a player "buys back" into the game, a musical stinger will play for all players, signalling this event.

There are also other clever uses of audio in *Dota 2* that highlight its importance in relation to gameplay. For example, one hero has a special ability called "Global Silence" that disables all enemies from casting spells. As well as its gameplay and visual effects, this ability causes almost all sounds to be turned down in volume to the extent that they are essentially inaudible. This ability therefore has gameplay ramifications in that it stops players from using sound cues to their advantage for the 4–6 seconds of its duration, a long time in gameplay terms given the speed at which action can happen.

Strife

Strife (S2 Games, in development) is a game similar to *Dota 2* which also takes a layered approach to its soundtrack. In a personal communication, sound designer

Stephen Baker of S2 Games explained how the music system of *Strife*, also known as the Conductor, works. There are a number of parameters that the music system uses to determine which layers of music to play. First, the location of the player's hero is used to choose between two different base layers of music depending on whether the hero is in a *lane* (where the main action happens) or in the *jungle* (the no man's land between lanes; see Figure 3.6). The next parameter is whether or not the player is in proximity to a *tower* (a defensive structure in the lane) which is taking damage. If this is the case, an additional percussive layer is added to the base lane layer to reflect the heightened importance of the event.

Figure 3.6 Layout of *Strife* level: lanes highlighted in blue; jungle areas highlighted in green.

The next parameter is whether or not there is any fighting between players happening. If the player comes within range of a certain number of enemy players, the current location base layer is replaced by the *teamfight* base layer. This new "more exciting base layer" (Baker, personal communication, 2013) reflects the risk of being in close proximity to enemy heroes. There are two further layers that can be added on top of the teamfight layer when either side initiates the fight. Depending on how many heroes on both sides are involved in the fight, either one or both of the layers will be added to reflect the scale and significance of the engagement.

There are two map objectives within *Strife* known as the "guardians" who can be defeated to gain a bonus for the team. When either of these mini-bosses is attacked, the music system creates "a more epic music track" (Baker, personal communication, 2013) by combining the teamfight base layer with either one of two special additional layers, tailored to the specific guardian. One of the bonuses unlocked by defeating one of the guardians is the aid of a giant gorilla named Krytos who adds his own music layer, featuring congas and other percussion, "to beef up that particular situation" (Baker, personal communication, 2013).

If a player is killed, they have to wait a certain length of time before they can re-spawn. During this time, "a more atmospheric music bed is introduced" (Baker, personal communication, 2013) and a timpani roll stinger plays to alert the player that they are about to revive in case they are looking elsewhere.

The soundtrack of *Strife* has the ability to reflect the meta gameplay progression that happens during the game due to the increase in power of each player. Once all of the external defences of either team's base have been destroyed, the original location base layers are replaced with, as Baker describes, "darker, more tension evoking versions of the same tracks" (personal communication, 2013). This change reflects the fact that the game is drawing to a conclusion. The final parameter that the music system uses is the proximity of players to either of the team's main base structures. If there are a certain number in close proximity to one of these, it is likely that one team is about to win; therefore, a completely new base attack/defend cue is played with further layers added as players attack the structure and it takes damage.

All of the cues are composed to the same tempo, essentially creating one huge composition, although the tracks will never all play together. The main cues cross-fade between one another in order to make major state changes, and the additional layers belonging to these cues also fade in and out. Both of these transitions use logarithmic volume changes.

Stephen Baker explained that much consideration had gone into the balance of the emotional and informational feedback of *Strife*'s music. Early on in its development, the music system had given far more gameplay relevant information to the player: for example, as Baker explained, if a player was in danger but did not realise it, the music could potentially inform them of this fact and allow them to respond accordingly. However, this was not the effect that the designers wanted from the music and therefore this aspect was curtailed. Despite this, Baker believes that there are still aspects of gameplay feedback alongside the emotional impact of the soundtrack:

> But even though the music is set up that way, there is a blend between emotional feedback and gameplay feedback. And it's because of that gray area that I believe the music has a positive effect on a player's performance. The music gives the player a dynamic that reaches low and high. It gives the player that much needed time to breathe when a fight has subsided. But then it ramps up to this epic cacophony of instrumentation that makes the player feel a heightened sense of attachment and involvement leading to better execution during a fight. (Baker, personal communication, 2013)

While the system no longer gives players information about unseen events, it does, however, still reflect the actions that are on-screen. Similar to the music in *Dota 2*, this doubling up of information in the visual and aural communication channels no doubt makes the various gameplay situations far more readable than would be the case without the music.

Multi-player

The three games discussed so far – *StarCraft II*, *Dota 2* and *Strife* – all put a heavy focus on multi-player gaming, and this can have a significant effect on the design of their music. First, multi-player games have vast replay value, meaning that they can be played over the course of months or years, and even the best music can become repetitive when listened to thousands of times. Because of this, many players choose to turn off the music of multi-player games.

It is also the case that many players choose to listen to their own music instead of the in-built soundtracks. *Dota 2* and *Strife* are primarily team games where communication, particularly over voice-chat software, is often present. This then becomes another reason for music to be turned off so that it does not mask teammates' communication. These aspects combined have created an expectation within many games, particularly ones that include multi-player, that music should not have any relevance to the information system of a game. There appears to be an attitude that games that include a competitive element almost by definition cannot have music that gives gameplay advantage because of the expectation of players that they can turn off the music.

Left 4 Dead 2

Left 4 Dead 2 (Valve, 2009), while not an RTS game, features a very interesting example of music being used for a clearly informative purpose. The game is a cooperative, first-person zombie shooter that features levels with randomly spawned (created) enemies and thus huge replayability. Being good at the game is less about memorising level structure and more about the skill of situation reading and execution.

Mike Morasky, composer at Valve, in a talk given at Steam Dev Days 2014 (Morasky, 2014), explained that they wanted *Left 4 Dead 2*'s soundtrack to add value to the experience so that players would be less inclined to turn the music off. They began experimenting by adding a musical theme to indicate when the most powerful enemy (a giant mutated zombie known as the Tank) had entered the level. This motif foreshadows the arrival of the Tank, as well as giving an indication of the direction it is coming from. Players found this information incredibly useful during testing, so much so that, as Morasky (2014) noted, when a bug within the music system caused the Tank theme not to play, there was an outcry from players who thought this feature, which they had come to rely on, had been removed.

The success of the Tank theme led the developers to add motifs to all of the other "special infected" enemies (mini-bosses), as well as when an ally becomes incapacitated and needs assistance. These small musical motifs are added on top of a bed of ambient music and warn the player of the presence of a "special infected" enemy before the player can see them. When the enemy finally attacks and is visible, the theme then turns into a fuller version that dominates the music. Interestingly, though, as Morasky (2014) noted, during testing, certain players reported that they did not notice these music cues, although when their play was observed, it became clear that they were subconsciously being affected by these musical motifs as they could be seen to back away or otherwise ready themselves for the "special infected" when the cues played but before the enemies became visible.

4

Methodology

Chapter 2 has discussed the vital role that game audio plays as part of the information system of a game, while Chapter 3 has explored examples of the audio in current RTS games, some of which include vertical layering music systems. In addition, the potential of incorporating music into a game's information system has been demonstrated by the example of *Left 4 Dead 2* (Valve, 2009).

This research has therefore designed a methodology in order to investigate and test further the hypothesis that a vertical layering soundtrack can be used as an integral part of a game's information system and in a way that has so far not been attempted within the field. A series of three purpose-built test levels were developed to act as both prototypes for the music implementation and, more importantly, empirical user tests of the systems and music. The tests were designed to see whether the participants could play the levels while actively listening to the music and thereby gaining information about the game from the soundtrack. To measure how well the players understood the information being conveyed, they were asked to make responses based on the information that they had potentially received. This methodology was designed to explore the concept of transmitting information via a vertical layering soundtrack and potentially highlight the strengths and weaknesses of the approach.

The initial intention was to use the *StarCraft II* Galaxy editor (a development tool for creating *StarCraft II* mods) as the platform for creating the test levels. There were thought to be many advantages to using the *StarCraft II* editor: for example, the test levels would be built within a pre-existing RTS game (arguably one of the most popular current RTS games), the levels could be published via Blizzard's own battle.net servers, as well as potentially being able to reach a large pre-existing online community of gamers. Unfortunately, due to limitations of access to low-level audio control, there were problems with synchronisation of multiple layers of music and so use of the *StarCraft II* Galaxy editor was ultimately abandoned. Nevertheless, the platform did serve as a useful prototyping tool for experimenting with early concepts quickly and efficiently.

After evaluating other potential game engines, such as UDK and CryEngine, Unity3D was chosen for the final test levels. As a full game engine, Unity3D does not have the technical limitations of the *StarCraft II* editor. It also meant that the test levels could be completely custom-built, which gave complete control over the testing environment, which brought many advantages. Previous researchers who have attempted to do user testing have generally relied on using pre-existing games or measuring the reaction of participants while watching film clips with different musical accompaniments (see, for example, Moffat and Kiegler, 2006). In Jørgensen's (2008b) "Left in the Dark" research, she relied on simply turning the sound on and off in *Warcraft III* (Blizzard, 2002) and *Hitman Contracts* (Io Interactive, 2004), in order to test players' experiences with and without sound and show the effect this had on their performance.

Anyone with access to a PC or Mac could download and run the test levels as there was no need to own an existing game. Likewise, players would not be required to attend a testing session in person or have the equipment brought to them, meaning that logistically the methodology was much simpler to organise. Most importantly, though, building the testing environment from scratch in Unity3D created the possibility to design exactly the right gameplay scenario, music system and testing mechanics for the needs of the research.

Before the individual test levels were designed, the minimum necessary components to construct an RTS game were developed within Unity3D. These components included the input and interface, including camera perspective and movement, as well as mouse selection, and player and computer-controlled units which would have the ability to move, attack, have health and be semi-autonomous, as well as simple enemy AI for the player to interact with. Considerable emphasis was put on creating the right look and feel to make the tests appear as close as possible to a real RTS game. Therefore, simple models were created for the various types of units and other structures, as well as graphics for the user interface (UI) and testing mechanics. While the first test level featured a music system programmed using Unity3D's inbuilt audio functionality, during the course of the research Firelight Technologies, makers of the audio middleware Fmod, released a plug-in for Unity that enabled users to integrate Fmod Studio into Unity projects. This was therefore used for the second and third test levels.

The aim was to test the levels on game players. Not only would they be able to provide qualitative feedback on how they found the experience, but also, more importantly, built into the levels were mechanics to record their performance during the tests, allowing for detailed quantitative data analysis. Each level featured UI buttons that the player used to respond to the soundtrack information, the results of which were recorded behind the scenes and used for later analysis.

A small group of 19 individuals took part in the research, some of whom were approached to participate due to their experience of playing RTS games and others volunteered after hearing about the research through word of mouth. It should be noted that some of the participants were known personally to the researcher, which could possibly have created a more positive attitude towards the research. This, however, would only affect the oral feedback given and not the quantitative test results, given that it was not deemed possible to *do better* in the tests because of any bias, but only potentially *try harder*. All participants were male, over 18 years old and came from a range of gaming backgrounds, which included varied RTS playing experience. The research was presented to the players under the title of "The RTS Soundtrack Experiment", and other than the fact that players were told that the research was investigating dynamic music in games and that they were required to play a short game level, all other information they received about the research was gained from within the test levels themselves.

The individual test levels were self-administered by the players. This aspect of the methodology had a number of advantages. It was hoped that this would enable more players to participate in the research. The fact that the administration of the tests did not have to happen all at one time and require the players to travel to the same location suited the iterative approach of the level development. If players were inconvenienced as little as possible, it was hoped that they would be more willing to participate in follow-up levels. Finally, having players administer the tests themselves in their own time meant that they would be playing the levels in a comfortable and natural environment, and this was likely to be where they usually play games.

Each test level collected a series of results from the player. These results represented how the player responded to the soundtrack in the level based on the information they had received and were used for quantitative analysis. The levels also collected data on other non-musical aspects of the gameplay for the purpose of creating comparisons between the two types of data. As well as this, those players who were willing to give feedback were interviewed about their thoughts and experiences with the soundtracks. Only a small number of players were able to give such feedback, and it was clear during the process that many players did not feel comfortable or able to talk to any extent about their experience. Despite this, the views gathered from the small number of participants who did answer questions supplements the main results with qualitative feedback.

It should be noted that the feedback mostly came from the players who did "well" in the tests. It is likely that these players were the most engaged with the research and this meant that they did better and were more willing to give feedback, although it is unclear whether players did well because they were engaged with the research or were engaged with the research because they did well. The interviews themselves were informally structured, with only a handful of predetermined general questions. These interviews took the form of remote Skype calls, carried out shortly after the player had finished the test. This was primarily so that the test was still fresh in the player's mind, but this had the disadvantage that there was no time to analyse the player's results and thus tailor questions specifically to that individual other than those questions that came up during conversation. In addition, given that players were interviewed one by one as the research was in progress, specific patterns in the data, which became apparent when all results had been analysed, were not able to be discussed with individual players.

Although there were many advantages to the methodology, a number of disadvantages should be considered. The method of testing and, in particular, the fact that tests could be self-administered meant that each player would be playing the test levels on different equipment. Each player could potentially have a very different experience of the test, and in particular the music, depending on their speakers or headphones. Allowing the players the freedom to take the test on their own and in an environment unknown to the researcher meant that there was potential for interruptions or other disturbances that might disrupt the flow of the test. It is possible that the disconnect between researcher and subject may have caused a lack of effort or commitment to the research, although there is no specific reason to believe that this was the case. Finally, it was also possible that the player might be tempted to re-take a test in order to get a "better" result. In order to minimise this, players were told that there was no such thing as a "bad" result and that the music was the aspect being tested, and not them.

All test levels used a simple but recognisable RTS game scenario which, with small variations, consisted of a player's base with a random number of enemies periodically spawning to attack it. The number of enemies provided an appropriate type of information to convey to the player through the layers of the soundtrack. During the tests, players were asked to respond to the music by clicking buttons to indicate their expectation of the number of attacking enemies.

The development of the test levels took an iterative approach. While the general progression between each level was planned to increase in complexity and difficulty, results and feedback from one level fed into the development of the next.

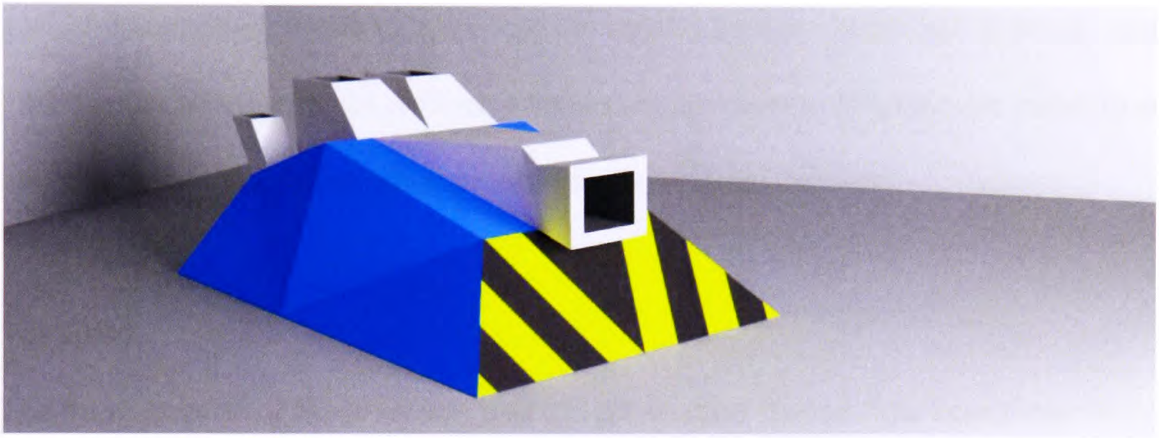
Level 1 was designed to explore whether players could listen to and distinguish between different layers of a soundtrack and whether certain aspects of musical composition worked better than others in terms of transmitting information to the player. Although this first level set out to look at something quite simple, it was necessary to establish early on whether players can listen to and interpret aspects of a soundtrack while they play before making the tests more complex.

Level 2 aimed to amplify the complexity of the test and the music system by increasing the number of layers in the music from four to six, including the base layer, and hoped to improve the accuracy of information transfer based on the findings of Level 1. In addition, the level also featured a further gameplay task to distract the player and simulate a more interactive experience.

Level 3 was designed to add further complexity to the soundtrack. Whereas in the first two levels, only one parameter (number of enemies) was being sonified in the music, this time the layers represented two different parameters: the musical layer itself took on a motivic characteristic and represented a *type* of enemy, while the volume of that layer represented the *number* of that type of enemy. A second distraction task was added to the game to allow for further comparison between soundtrack understanding and gameplay proficiency. Each level is discussed in detail in the following three chapters (Chapters 5–7).

5

Level 1: A Basic Vertical Layering Soundtrack



DESIGN

In order to begin to explore the hypothesis that a vertical layering soundtrack can be used as part of the information system of an RTS game, Level 1 was designed to be a simple test to see whether players could identify individual layers of a soundtrack and relate these layers to information that they were told they represented. The number of enemies attacking the player's base at a given time was used as the information to be transmitted by the soundtrack.

The design of Level 1 was deliberately kept simple. It was important to confirm during this early stage of testing that players can interact with the game and listen to the soundtrack at the same time in this particular context. This needed to be determined before the tests became more complex. If this could be established, then later levels would be able to explore in greater detail the idea of vertical layering conveying information.

The soundtrack of Level 1 consisted of four layers: a base layer that played continuously and three further layers that would be added on top of one another to represent different numbers of attacking enemies. The aim was for each layer to increase the intensity and danger of the music overall. Each layer focused on using different parameters of music to achieve this, and therefore, in addition to the primary goal of determining whether players can distinguish between individual layers to gain information, this test might indicate whether certain aspects of music are better than others for conveying information in a vertical layering soundtrack.

The Level

The basic premise of the level was that the player must defend their base from waves of attacking enemies (see Figure 5.1). This set up a very simple yet recognisable gameplay scenario that features in many games, particularly of the RTS genre, and therefore would be familiar to many of the players. The enemies that attacked the player's base did so in groups of varying size which was randomly generated. As well as being a common feature in games (randomly generating parts of the gameplay adds variation and increases replay value), this aspect of the design created a piece of information that could be transmitted to the player through the soundtrack.

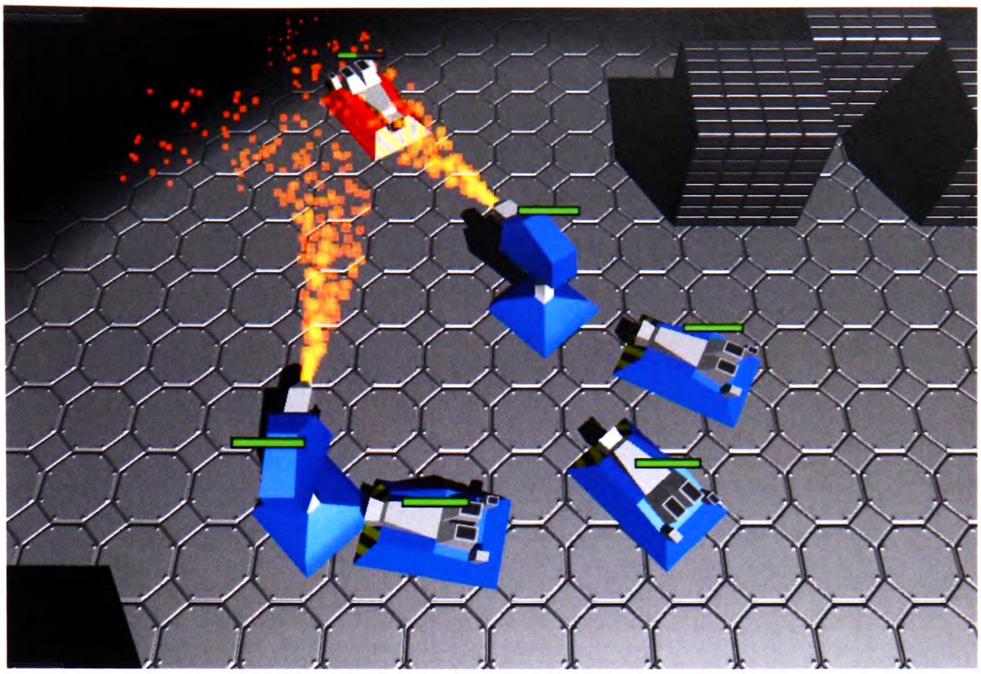


Figure 5.1 Flame turrets defending the player's base from enemies in Level 1.

The emphasis in this first test was on simplicity, so it was important not to overwhelm the player. Although the player still had to control their units and push buttons to respond to the soundtrack, the game balance of the level meant that there was little actual danger to the player's base. Establishing that this method of testing worked and that a player can listen, understand and respond to the soundtrack while playing were the most important aspects of the first test level.

The Testing Mechanics

Integrated into the level were all of the mechanics that enabled the level to be a self-administered test. The test began with a simple start screen (Figure 5.2) which featured a looping piece of music, serving as an opportunity for the player to ready themselves for the start of the test and set their audio volume.

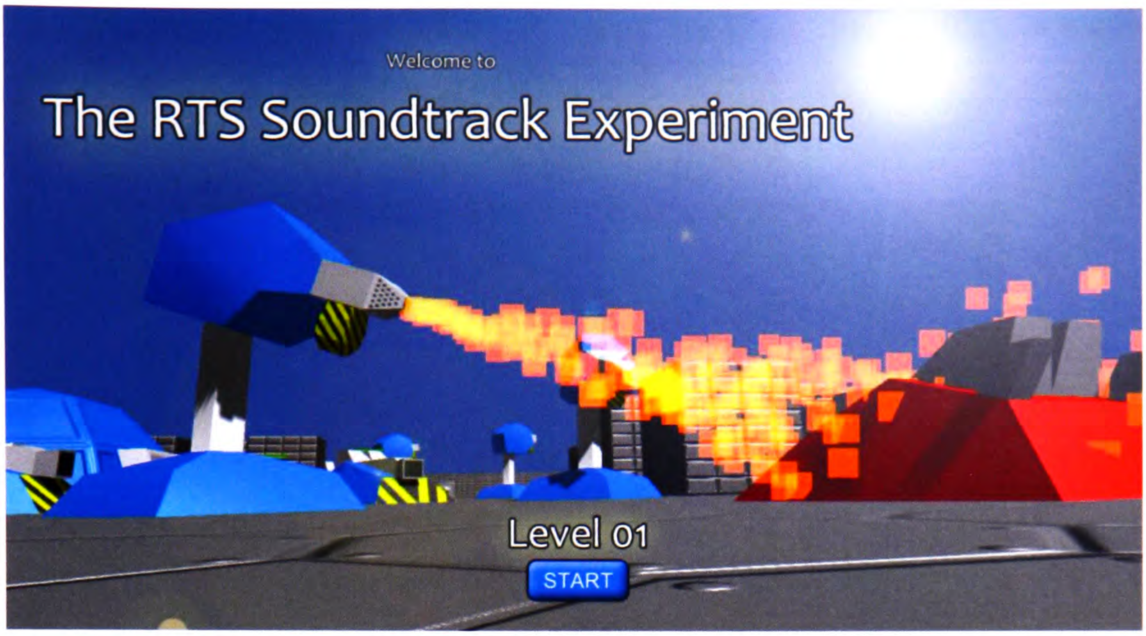


Figure 5.2 Start screen of Level 1.

When the player pressed the start button, the main level loaded and the test began with a tutorial section that introduced the level as well as explaining everything the player needed to know in order to complete the test (Figure 5.3; see Appendix 1 for a transcript of the tutorial). It was explained to the player that this was a study of how well dynamic music can aid a player during gameplay, although the details of the research were kept to a minimum so as not to influence the player or create any particular expectations. The basic controls, gameplay and testing mechanics, as well as the specific response buttons, were all explained to them.

The participants were played excerpts of the music so that they could have an opportunity to familiarise themselves with the individual layers of the soundtrack. This was to simulate a scenario where the player had some familiarity with the game and therefore had some understanding of its aural information. The idea was not to test what the player thought the music represented in terms of their own interpretation, but whether they could remember and associate the music with the learned meaning.

Finally, they were given instructions on how to submit their results and then the test began.

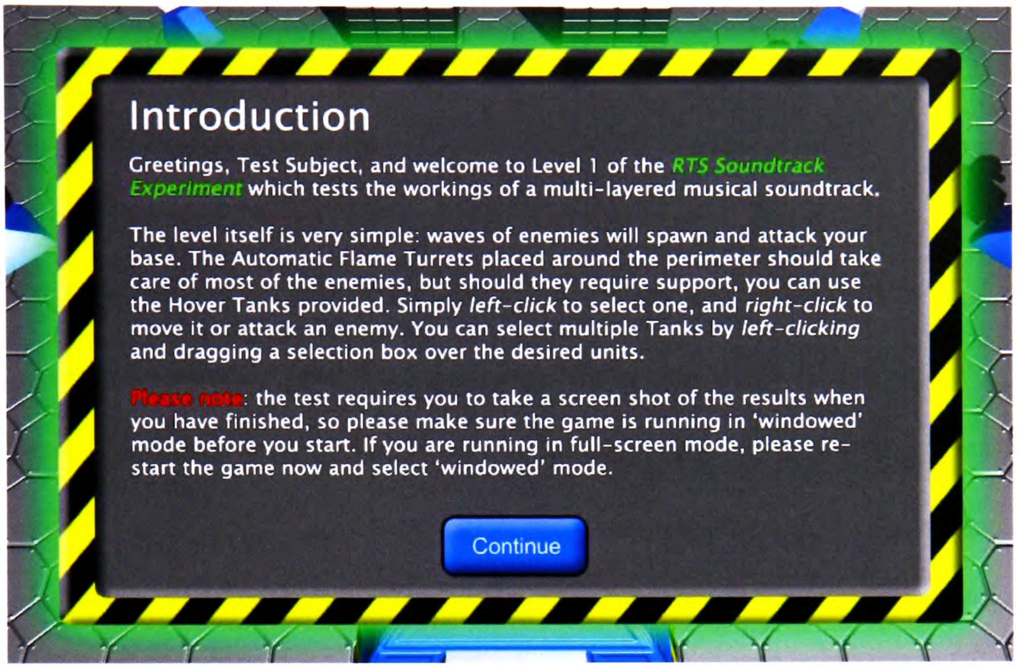


Figure 5.3 Extract from Level 1 tutorial.

As well as being given in written form, the entire tutorial was narrated. As there was a considerable amount of information to convey to the player at the outset, it was deemed important not to put off the player by a large amount of text to read. An auto-tuned effect was applied to the narration to create a robotic-like voice. This was initially done simply to match the sci-fi-esque aesthetic of the level, but it unintentionally created another effect. The voice, along with the fact that the level is in the form of a "test", produced a scenario slightly reminiscent of Valve's *Portal* franchise, a game where the player is forced to work through a series of physics puzzles by a rogue computer AI. This took the emphasis away from it being part of a research project, which may have put the player slightly more at ease and made the experience feel more "game like".

The Gameplay

It was decided that ten waves of enemies would be a reasonable number to generate sufficient data to see any trends in player responses, but not be too many to become tedious for the player. This meant ten individual responses for each player to make and to be analysed in the results.

When a wave of enemies spawned, the player received an on-screen notification and a visual countdown timer started. The enemies then moved slowly towards the base, hidden from the player in the *fog of war* (see Chapter 3). During this time, the music changed depending on how many enemies had spawned and the player was instructed to use the time before the enemies arrived at their base to make a prediction of how many enemies were about to attack. This information they received about numbers of attacking enemies was therefore only conveyed through the soundtrack and was not available visually: the enemies were completely hidden. If this were a real game scenario, it might be imagined that the player could then use this information to ready their defences. Although the test was essentially just limited to players receiving the information, they did have to act on it, not in terms of gameplay, but by responding through pressing interface buttons.

The player made their prediction by pressing one of the three buttons at the bottom of the screen (see Figure 5.4). In this first level, the possible predictions were groups of 1–3, 4–6 or 7–9 enemies. Once the player pressed a button, the result was stored and all of the buttons then disappeared, making it impossible to make more than one prediction per wave. Likewise, the prediction buttons disappeared before the enemies arrived, meaning that the player could not simply wait and count them. The on-screen notification then changed to inform the player that they were now waiting for the next wave and the music returned to its normal state.

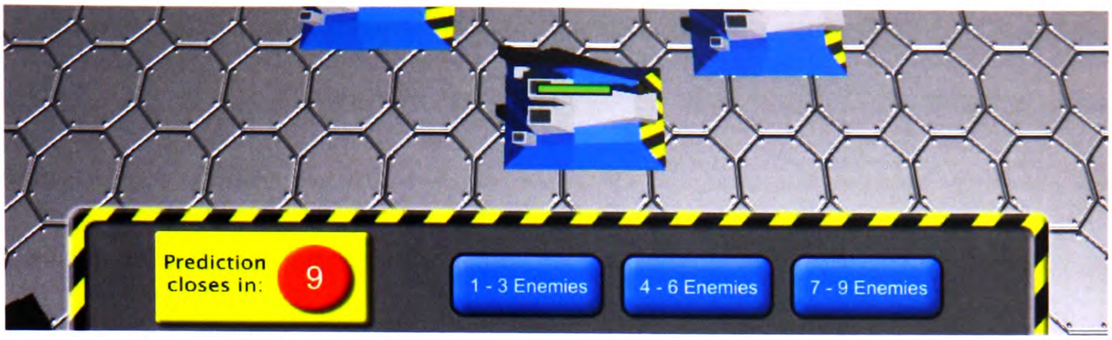


Figure 5.4 Level 1 prediction buttons.

The fact that the number of enemies comprising each wave was randomly generated meant that no two tests would be the same. This also alleviated other potentially negative aspects of having the same wave-spawn sequence for all players. For example, one aim of the analysis was to see whether there was any improvement in player prediction over the course of the test, meaning that players were learning the music. If certain music layers, and therefore wave groups, were harder to predict, without random spawning the results over time might reflect the difficulty of these layers/wave groups and not whether players were improving or not.

Once the ten waves of enemies had been completed, a final dialogue panel appeared, thanking the player for their participation, along with a "fetch results" button, which, when pressed, displayed the results of the test. The player was asked to take a screenshot of this panel and email it to the address displayed below the results.

Data Storage

During the test, the results were stored by two integer arrays, essentially lists of numbers. The first list stored what size of wave spawned and the second stored what size the player predicted. When the wave spawner selected a wave size to spawn, as well as literally creating the enemies, it sent a message to the part of the code handling the results storage telling it what wave had been spawned. It did this by sending a single-digit number which corresponded to the wave size: 1–3 enemies sent a "1", 4–6 a "2", and 7–9 a "3". This number was then stored in the array at the index that corresponded to the wave number. The same process happened when the player made a prediction, only this time the button they pressed determined what number was sent to be stored.

When the results were printed out for the player at the end of the test, the code simply went through both arrays and printed the number from each corresponding index next to one another which then essentially created a row of ten two-digit numbers (see Figure 5.5). So, for example, if wave 1 had been 7–9 enemies and the player had predicted correctly, the first result would have read 33. With this method of storing and displaying the results, one could quickly see at a glance how well the player had done – multiples of 11 showed correct predictions. There was one other advantage of storing the results in this way. As the default value for indexes in an integer array is zero, if the player failed to make a prediction, no results would be stored and the value at that index in the array would remain at zero. Therefore, any results that were multiples of 10 showed that the player did not make a prediction.

Figure 5.5 Level 1 results screen.

The Compositional Process

In order to match the low-poly and sci-fi-esque aesthetic of the visual aspects of the level, the music was created using an electronic and chiptune-style instrumentation. The music was composed and arranged in four individual layers, each consisting of multiple instruments/VST samples and so should not be thought of as distinct instrument lines or parts (see DVD item 01 for an extract of Level 1 music). For clarity of discussion, the musical layers will be referred to in a similar way to floors of a building – the "ground" floor being the base layer, followed by layers 1, 2 and 3 above. It makes sense to refer to the lowest layer of music in a slightly different way from the others because this layer was constant and not adjusted by the music system.

The base layer consisted of three instrument parts, all electronic synthesisers: a bass, a pizzicato and a higher soprano part (see notation in Figure 5.6 and DVD item 02).

Figure 5.6 Level 1: extract from base layer.

Layer 1 added an arpeggio pattern using the same pizzicato instrument featured in the base layer, turning this slow, plodding line into a fuller, more energetic part (see notation in Figure 5.7 and DVD item 03). This technique could potentially be termed *instrument stacking*, where the same instrument is used in two or more layers which transforms one instrument part into something new by adding further notes to the original part. This technique could most obviously be used to make a slow part faster. While its use here adds further polyphony to the pizzicato part, this technique might have a different effect with percussion instruments that have a fast attack and decay, enabling more notes to be added without any unrealistic overlap. This layer also adds alto and tenor synthesisers, adding further depth and movement to the music. It was hoped that these three parts would combine to represent the increased tension of a small number of enemies attacking the player's base.

Figure 5.7 Level 1: extract from layer 1.

Layer 2 introduced cymbals as well as two different drum-like instruments, which aimed to add further tension and a feeling of danger by playing against some of the rhythms in the lower layers (see notation in Figure 5.8 and DVD item 04).

Figure 5.8 Level 1: extract from layer 2.

Layer 3 mostly added a melodic line played in octaves on a lead synthesiser. It was hoped that this high instrument part would complete the music with a full, main melody line. Timpani and a bass drum were also included to add further impact to layer 3 by accentuating the first beats of the bars (see notation in Figure 5.9 and DVD item 05).

Figure 5.9 Level 1: extract from layer 3.

The idea behind this composition was to represent numbers of enemies through a feeling of increased intensity and danger with each layer added. The compositional challenge, therefore, came from the fact that layers were not removed to make room for others, but just added on top of one another. Obviously, in the case of this level, when the enemies were destroyed and the attack wave stopped, the music returned back to normal and the additional layers were removed. But there would never be an instance when a layer would be playing without the layers beneath it also playing. Because of this, the layers had to act as building blocks on top of one another, each one changing the musical feel from one state to another, and ideally seamlessly slotting into the layers below. Perhaps the best example of this can be heard in the interplay between the base layer and layer 1, where the additional notes of the second pizzicato line aim to integrate seamlessly into the base layer almost as if the part had moved on to a new section if it had been a linear piece of music.

It should be noted that, in the gameplay scenario as designed, no other sound effects are played during the time that the player uses to listen to the soundtrack for information. Only when the enemies arrive are sounds such as flame throwers, lasers and explosions heard. Because of this, in a way, the test level sets up a *perfect* listening situation. In a real game, depending on the specific situation, there may well be numerous sounds competing for the player's attention.

The Music System

The music system was built up of a hierarchy of Unity *GameObjects* that communicated with each other to perform the dynamic music functionality. Starting from the bottom up, the following made up the music hierarchy. The very lowest part of the music hierarchy was the *LayerObject*, which was responsible for controlling the individual audio files. There was therefore one LayerObject per track of music within the system. It contained functions to start, stop and control the volume of the audio file that was contained in its layer.

Next in the hierarchy was the *MultitrackObject*. This was a single object that controlled all of the LayerObjects below it. It acted as a middleman between the upper and lower parts of the hierarchy. It sent specific information, such as which layer needed to change volume and what volume it needed to change to, down the hierarchy.

At the top of the hierarchy was the main *MusicObject*, which was the overall music controller for the level. It contained a tailor-made script for Level 1. This exemplified why the MultitrackObject was so important: it, as well as the

LayerObjects, were general purpose, whereas the main MusicObject was specifically designed for a particular level. In this level, the MusicObject's main role was to count how many enemies were in the level and communicate this to the MultitrackObject. Its main function, "CountEnemies", did just that. If it counted 1–3, it told the MultitrackObject to turn up the volume of layer 1. If it counted 4–6 enemies, it told the MultitrackObject to turn up the volumes of layers 1 and 2, and, finally, if it counted 7–9 enemies, it told the MultitrackObject to turn up the volume of all three layers.

Finally, there was the *MetaObject*. Whilst not strictly part of the music system, as it contained functionality that controlled many aspects of the level, it communicated with the main MusicObject. As the music system only needed to react to the enemy count at two distinct times (when they were created and when they were destroyed), the MetaObject, which spawned the enemies, was the best place to communicate this to the main MusicObject. As well as this, each enemy, when it was created, made itself known to the system so that when it died the MetaObject told the main MusicObject to recount how many enemies were in the level.

Potential Findings

There were a number of predictions made about how the players would respond to the music of this first level. First, with regards to layer 1, it was thought that the difference between the base layer and layer 1 might be too subtle for the player to detect. It was not a particularly noticeable volume change, and especially because the additional base pattern subtly slotted in with the existing line, the player might not notice this transition.

The addition of layer 2, the percussive layer, was arguably the most obvious change. Players would be likely to notice this change more than the others and this layer might potentially have the highest correct prediction rate. Finally, it was possible that layer 3, the melodic layer, might not have the same level of impact as layer 2, and therefore might not be as noticeable a change to the player. However, it was the layer containing the highest pitch/frequency instrument, and therefore would be likely to stand out regardless of its likelihood of being overshadowed by layer 2. While it was likely to have a fairly high correct prediction rate, it was possible that this would not be as high as layer 2.

It should be noted that, while the discussion here and in Chapters 6 and 7 talks in terms of the individual musical layers, the nature of the music system means that, while referring to an individual layer, it is often implied that this includes all of the layers below it.

Potential Issues

One aspect of the test that could potentially have been an issue was the time allotted for players to make their predictions, and, more notably, the time the player was given before the very first prediction. It was believed that the player might not have enough time to adjust to the testing situation before they were asked to make their first prediction. Therefore, it was possible that extra scrutiny might need to be given to first predictions as this aspect of the test could have a negative impact on the accuracy of the results because of the nature of the test rather than the player's understanding of the soundtrack.

Another related issue was how the particular layers of music fade in during the time the player is asked to make their prediction: it might be potentially confusing for the player to have the music change while they were trying to understand the soundtrack. In order to lessen the impact of this, once the enemies had spawned, they were ordered to wait for a few seconds. This gave the music time to adjust, and only when the enemies began to move towards the base was the player given the notification to make their prediction.

One final issue could have occurred due to the nature of the randomly generated wave spawning. It was entirely possible that the same wave size could spawn many times in a row, which would have a negative impact on the results. For example, if a 7–9 group spawned for the first three waves, the player might think something had gone wrong with the test or that they could not tell the difference between the layers, which could confuse the test subject and influence their decisions about later waves regardless of which wave sizes spawned afterwards. Even if this did not influence the individual player, the results as a whole could be affected. If player A got too many 1–3 enemy waves and was good at recognising that layer, while player B got too many 4–6 enemy waves and was also good at hearing that layer, but both players were not good at recognising the opposite layers, this would make it look as though the music for 1–3 and 4–6 was more effective than it actually was.

For a video compilation of extracts recorded from Level 1, see DVD item 06.

To play Level 1, see DVD Level 1: test executable folder.

RESULTS

In all, 17 players participated in this first test, and Table 5.1 gives an overview of each player's results, displayed in the same way as they were recorded during the test (as explained above). The result is a two-digit number, the first digit representing what size of wave spawned and the second digit representing what the player predicted: a "1" representing 1–3 enemies, a "2" representing 4–6 enemies and a "3" representing 7–9 enemies.

Table 5.1 Each player's results for Level 1

	Wave 1	Wave 2	Wave 3	Wave 4	Wave 5	Wave 6	Wave 7	Wave 8	Wave 9	Wave 10
Player A	11	32	11	23	33	22	11	33	22	22
Player B	20	22	33	22	11	22	11	33	11	22
Player C	32	33	22	33	22	11	11	22	33	22
Player D	22	11	22	22	11	11	11	33	22	12
Player E	22	11	30	11	22	33	32	32	11	11
Player F	33	23	11	22	23	11	33	11	22	23
Player G	31	22	11	22	22	11	33	33	33	22
Player H	11	11	33	22	11	32	11	33	11	11
Player I	12	22	11	32	22	22	11	22	10	11
Player K	33	21	32	21	11	33	11	23	11	22
Player L	33	11	11	33	22	11	22	33	11	22
Player N	20	32	21	32	32	11	11	22	23	22
Player O	21	11	12	22	33	11	30	22	33	23
Player P	33	31	11	11	11	22	33	22	11	11
Player Q	11	22	22	22	33	11	11	33	23	30
Player R	12	21	22	11	22	22	22	11	22	33
Player S	23	20	22	33	23	23	11	11	11	33

For example, Player A's wave 5 result shows that 7–9 enemies spawned and he correctly predicted 7–9 enemies. Similarly, in terms of the music system, this could be thought of as layer 3 played and he chose layer 3. Because of this, and to simplify the discussion, from now on the results will be referred to as musical layers 1, 2 and 3. Although the players were predicting numbers of enemies, as the focus here is on the music it makes most sense to talk in terms of musical layers.

Which Was the Most Successful Layer?

This section looks at which of the musical layers was the most "successful"; that is, which had the highest correct prediction rate. Table 5.2 displays how many of each layer the players predicted correctly out of the total times played.

Table 5.2 Total correct predictions per layer in Level 1

	Layer 1		Layer 2		Layer 3	
	Predictions	Spawns	Predictions	Spawns	Predictions	Spawns
Player A	3	3	3	4	2	3
Player B	3	3	4	5	2	2
Player C	2	2	4	4	3	4
Player D	4	5	4	4	1	1
Player E	4	4	2	2	1	4
Player F	3	3	2	5	2	2
Player G	2	2	4	4	3	4
Player H	6	6	1	1	2	3
Player I	3	5	4	4	0	1
Player K	3	3	1	4	2	3
Player L	4	4	3	3	3	3
Player N	2	2	2	5	0	3
Player O	2	3	2	4	2	3
Player P	5	5	2	2	2	3
Player Q	3	3	3	4	2	3
Player R	2	3	5	6	1	1
Player S	3	3	1	5	2	2
Total	54	59	47	66	30	45
	91.50%		71.20%		66.70%	

First of all, we can see from the results how many of each individual layer was played. One of the original concerns, due to the random nature of the wave spawning, was that there might be an uneven distribution of results for the three wave sizes. However, for the 17 participants, and therefore 170 individual results, there was a reasonably even spread of wave sizes: 59 for layer 1, 66 for layer 2 and 45 for layer 3.

Next, looking at the total correct prediction percentages, layer 1 was identified correctly 91.5% of the time. This was surprisingly high, as it was originally thought that this layer might have the lowest correct prediction rate, given the subtlety of the change between the base layer and layer 1. Layer 2, however, saw a lower correct prediction rate at 71.2%. It was originally thought that this layer would have the highest prediction rate, given the nature of the percussion being a fairly prominent addition to the music. Finally, layer 3 saw the lowest correct prediction rate, with 66.7% of predictions correct. It should be noted that this should not necessarily be viewed as a low correct prediction rate, but only comparatively so when viewed against the high figure for layer 1. Originally, it was thought that layer 3 would have a slightly lower prediction rate than layer 2, and comparing those percentages in isolation, this is the case. Overall, though, given that, for each individual prediction, there was essentially a 33% chance for the players simply to guess correctly, the fact that the prediction rates for all three layers are much higher than this might suggest that players are, to some degree, recognising individual layers and thus receiving the information about enemy wave sizes correctly.

There is an obvious pattern in the results, with layer 1 being the most successfully predicted and layer 3 being the least. While the initial thought was that certain musical aspects would be more prominent, and thus increase their chance of being recognised and conveying information to the player, this was perhaps not the case. It is therefore likely that there is another, perhaps simpler, explanation for the pattern of results.

It is possible that this decreasing pattern of correct predictions has simply been caused by the fact that, as more layers are added, it becomes increasingly hard for the player to distinguish between them. With each layer comes more complexity in the music, and while this achieves the goal of conveying the musical representation of the gameplay information by increasing the intensity and danger of the music, it also comes at the price of making the music harder to interpret.

This would explain why the initial idea that layer 1 would be the least easily recognised was wrong. In fact, despite the subtle transition between pizzicato parts, there is very little else happening in the music at this time. Likewise, because of the dominance of the percussion parts added with layer 2, the players may have had difficulty identifying the melody line added in layer 3, despite its high-pitched nature.

This theorising is based on the limited amount of information gathered from Level 1, so a target for the next level was to try to get a more even correct prediction rate from all of the layers, or at least explore the potential limits of how many layers and what degree of musical complexity players can interpret during a play session.

Confusion between Layers

While much can be learned by looking at the percentages of correct predictions for each layer, it is also interesting to see how often layers were confused for other layers. So, for example, how many times was layer 1 confused with layer 2? For the sake of clarity during this discussion, the layers will be abbreviated to 1s, 2s and 3s (for example, layer 1 predicted as layer 2 would read "1s as 2s").

Table 5.3 displays how many times each layer was mistaken for another as a percentage of the total number of times the actual layer was heard. As shown, "1s as 2s", and its counterpart "2s as 1s", not surprisingly had a very similar confusion percentage: 6.8% and 7.6%. It is also a low percentage, meaning that these two were not often confused. Layer 1 in itself was predicted 91.5% correctly; therefore, it makes sense that layer 1 will not often be confused with any other layer.

Table 5.3 Confusion between layers in Level 1

Layer Predicted \ Layer Played	1	2	3
1		7.6%	4.4%
2	6.8%		22.2%
3	0.0%	16.7%	

The confusion of "1s as 3s" as well as "3s as 1s" had an even lower percentage rate: 0% and 4.4% respectively. Obviously, the musical difference between layer 1 and layer 3 is quite large and this is likely to account for why their confusion rate is so low. It is interesting, though, that these two percentages are not the same. However, when looking at the raw numbers, the 4.4% confusion rate for "3s as 1s" is caused by only two incorrect predictions. As mentioned in the Design section above, one worry with the style of testing was that players might not be prepared for the first wave of enemies and that early predictions might possibly be less accurate than later ones (once the player had become familiar with the testing scenario). Interestingly, as can be seen in Table 5.1, both of these "3s as 1s" came in the first two waves. While this is not a reason to discount any results, it is a possible explanation for why they appear.

Finally, "2s as 3s" and "3s as 2s" saw a higher rate of confusion: 16.7% and 22.2% respectively. While these percentages are slightly different from one another, it is worth noting that they were identical up until the last participant, Player N, who had three incorrect predictions of "3s as 2s" (see Table 5.1). Despite this, the numbers are still similar enough to see that the confusion between the two layers essentially worked both ways.

It is clear, then, that layers 2 and 3 had the highest level of confusion, and therefore were possibly not sufficiently different to distinguish between them. This could potentially back up the point made earlier that it is possible that the additional melody added in layer 3 does not do enough to change the perception of the music. Or, put another way, the musical representation of information does not change enough by the addition of layer 3. Similarly, it could be argued that layer 2 adds too much to the music in comparison to layers 1 and 3, and does not leave enough "intensity" for the final layer.

Player Performance over Time

Looking at whether the players predicted correctly or incorrectly for each wave, regardless of the actual wave size that spawned, reveals how the players as a whole performed in their predictions over the course of the test. This might indicate whether the players were able to become more familiar with the soundtrack, even learning to recognise individual layers, as they progressed through the test, even if they could not accurately recall them from the tutorial examples.

Figure 5.10 shows the number of correct and incorrect predictions per wave and displays a clear pattern in the players' performance over time. Starting at wave 1, which had nine correct predictions, this then increased all the way to wave 7 which had 16 correct predictions. Interestingly, though, after wave 7 the opposite trend occurs, with the number of correct predictions decreasing to the final wave 10, which contained 13 correct predictions. At first this might appear odd, given the assumption that players would improve over time because of potentially becoming more familiar with the music and more comfortable with the test itself, thus improving their ability to respond to the music correctly. If this were the case, it might be expected that the number of correct responses would increase all the way up to the last wave.

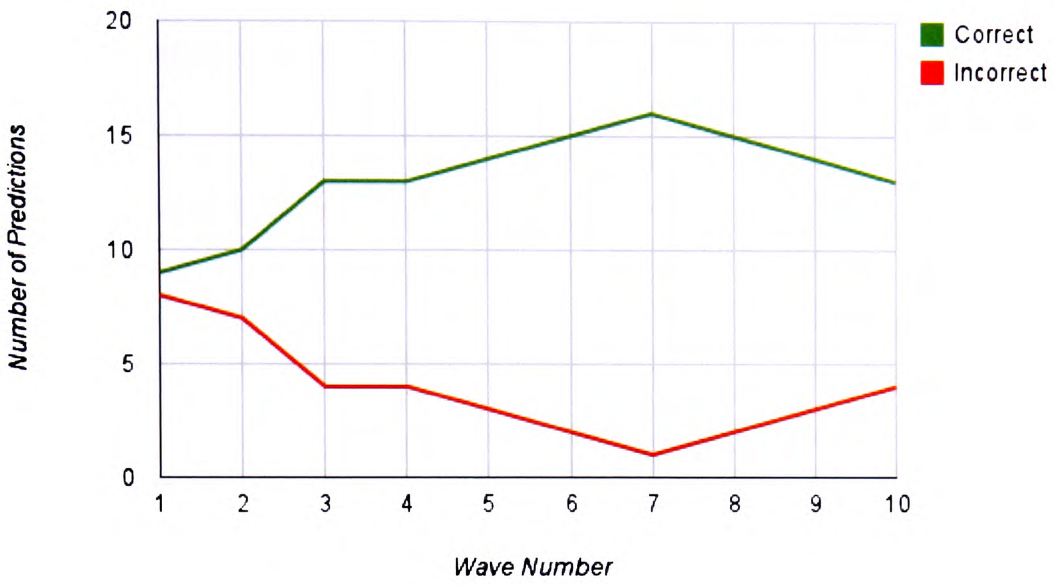


Figure 5.10 Player performance over time in Level 1.

As noted earlier, one of the initial concerns with Level 1, due to the nature of the random spawning, was that which wave size spawned at each particular wave number could potentially have had an adverse effect on the results. For example, as shown earlier, layer 1 had a much higher prediction accuracy than layers 2 and 3. Therefore, if the majority of layer 1 results came from around wave 7, where the highest response accuracy occurred, it may not have been the case that the player was learning the music and becoming more proficient as the test progressed, but merely because the most successful musical layer was being heard more often during these waves.

Table 5.4 displays the total number of each wave size that spawned for each wave. This shows any particular areas where an unbalanced number of one particular wave size spawned and thus possibly affected the results. In wave 1, layer 1 was heard five times, layer 2 six times and layer 3 also six times. This could be considered the most balanced distribution of wave size spawns: with 17 total respondents, 5/6/6 is as even as it can be. Anything outside these bounds, then, can be considered an influence in favour of the largest number. So, for example, in wave 2, the distribution is 5/8/4. This wave, therefore, is weighted towards layer 2. These potential negative influences have been highlighted in Table 5.4.

Table 5.4 Potential wave-type influence in Level 1

	Wave 1	Wave 2	Wave 3	Wave 4	Wave 5	Wave 6	Wave 7	Wave 8	Wave 9	Wave 10
Layer 1	5	5	7	3	5	8	10	3	8	5
Layer 2	6	8	6	9	8	6	2	6	6	9
Layer 3	6	4	4	5	4	3	5	8	3	3

It would make sense, then, to look primarily at the influence of layer 1, which had the highest accuracy of prediction rate, at 91.5%. The influence of layer 1 does to some degree correlate with the overall accuracy of predictions, with waves 3, 6, 7 and 9 all being influenced by layer 1. As the peak of correct predictions, wave 7 is of particular significance as this wave featured the largest wave size influence of all: 10 out of the 17 spawns were of layer 1. This explanation does not fit, though, when wave 6 is compared with wave 8. Wave 6 features a high influence from layer 1 and wave 8 features a high influence from layer 3. But the overall prediction accuracy of these two waves is identical (see Figure 5.10). While it is likely that the wave size is slightly influencing the results, it may not be the primary cause of the pattern.

It is still possible, then, that the overall pattern of prediction accuracy is being caused by the simple fact that the players become more familiar with the music and improve their predictions over the course of the test. The player starts the test only briefly hearing extracts of the music but, during the course of the level, they hear more of the music and become more comfortable with the test in general. Perhaps more importantly, throughout the course of the level the player is able to build up more and more comparisons between each wave size and the music they hear. When a wave arrives and it does not contain the number of enemies they thought it did, they can re-evaluate their understanding of that particular musical layer and this potentially explains the increase in accuracy of predictions. An example of this can be seen in Player K's results (see Table 5.1). In wave 2, layer 2 is heard but he predicts layer 1; then the same thing happens again in wave 4. At this point, he has not actually heard a layer 1 on its own. Then, finally, in wave 5, a layer 1 plays and he predicts correctly, potentially realising that he had not heard layer 1 up until this point.

The tail-off in prediction accuracy could potentially be explained by the fact that the test lasts over ten minutes, and it is possible that the players could be becoming tired or even bored with the testing. Another possible explanation could be that if the players find themselves predicting most of the waves correctly (and there is evidence to show prediction rates increase towards the latter half of the test), they could become complacent or lose focus and not perform as well as they might have done.

FEEDBACK

Confusion between Layer 2 and Layer 3

An issue common to many of the players was the confusion between layers 2 and 3. Players reported that they could recognise layer 1 without much problem, and the prominent percussion in layer 2 was also something that a number of players stated they could recognise. However, many players reported that they could not distinguish much difference between layers 2 and 3. This was exemplified by the feedback given by Player H, who seemed slightly confused and worried that he had not heard layer 3 at all. As noted previously, there was a concern, due to the random nature of the wave spawning, that a player might potentially not hear one of the layers. However, Player H's results show that he did in fact have a reasonably good spread of wave types, and while he claimed to have relied on "guessing" many of his layer 3 responses, he in fact only got one prediction wrong. It is, of course, possible that he did purely guess those responses, but it is not completely unfeasible that something in the addition of layer 3 influenced his "guesses", causing him to get most of them right.

Simply adding layer 3 to the music increases the complexity and general loudness of the music, but the elements of layer 3 might not have been enough for players to hear it as the top layer. Traditionally, the melody of a piece of music will often be one of the highest pitched parts and will sit on top of the rest of the music. This was why the melody part was included in layer 3. Given the confusion players felt between layers 2 and 3, it is possible that having the melody as the final layer was counterproductive. Like Player H, Player E did not think that there was enough of a difference between layers 2 and 3. He thought that it could be that the melody part somewhat covered up layer 2, which might have made the percussion sound less prominent. Thus, while the players could still hear the percussion, it did not necessarily sound as if the music, overall, had become louder. As the layers were meant to be informing the players of increasing danger, the fact that the melody may be smoothing over the fast rhythm of the percussion was actually counter to the original goal.

Grandeur of the Music

As an extension of the previous point, many players reported that they thought layer 3 should have been more intense, louder and dangerous sounding. Player K, in particular, stated that he thought the music for layer 3, which represented 7–9 enemies, should be more "grand" in order to "spur the player on". As noted in the Design section above, the actual difficulty of this level was very low: it was almost impossible for the player's base to be destroyed and for them to "lose". Seeing how well the player could do in defending the base was not the point of this level. Player K's feedback was particularly interesting as he clearly entered into the spirit of the game and used his imagination to envisage the scenario as an epic struggle, while in reality there was very little danger at all. The difference in threat between the three different groups of enemies was not particularly large, but potentially because of players' expectations, based on prior gaming experiences, the most dangerous wave in this level did not seem threatening enough. Likewise the music, regardless of whether it matched the literal danger of the scenario, did not match player expectation.

A similar feeling was also reported by Player E. As well as the confusion between layers 2 and 3, because of the failure of the melody to add enough extra intensity to the music, Player E stated that, although he chose layer 3 several times, he was not confident in his predictions and was not sure whether he should be waiting for an even more intense layer to play.

The Feel of the Music

Player E stated that he did not try to memorise the music at the tutorial stage: he rather thought that that approach would be "cheating" in the test. Technically, the player was not asked in the tutorial to *memorise* the music, but just given examples of the base layer and the three layers that they would hear in the test. This attitude seemed to be common amongst many players, who approached the task, and listening to the soundtrack, in the same way they might while playing any other game. Rather than *learning* the music in the tutorial and trying to recognise it later in the test, as was expected, it seems that many players just relied on the *feeling* of the music itself to convey the information about numbers of attacking enemies.

CONCLUSION

The initial concept for the level was to explore whether players can identify individual layers within a vertical layering soundtrack and gain gameplay information from this: in this case, numbers of enemies attacking their base. While this test has shown that to some degree this can be the case, it has arguably, more importantly, highlighted the fact that a crucial aspect of vertical layering is to pay attention to accurately mapping the information to the music.

As discussed in Chapter 2, it is believed that information in the form of continuous messages suits vertical layering as a vehicle for information transfer because of the ability to add and subtract layers as the information changes. However, it is important to consider how the information can change or how it is structured. Is it a constant progression? Does it change in steps or is it structured by a certain metric? If you could visualise the information on a graph, how would it look: linear or logarithmic, for example?

In the case of Level 1, the information about how many enemies are attacking the player's base in the current wave is structured in steps (the three wave types): 1–3, 4–6 and 7–9 enemies. This could be considered as a linear progression as the difference from one step to the next is even. When compared with the musical progression in the four layers of the soundtrack, however, it is clear that the two are mismatched, and the music does not progress *evenly* in the same way as the information being represented does. Using danger as a measure, Figure 5.11

represents a mock-up of this mismatch. While the intensity due to the danger of the increasing number of enemies progresses in a linear fashion, the music acts differently. Layer 1 adds to the intensity slightly by increasing the pace of the pizzicato instrument. With layer 2 added, the intensity increases sharply due to the large amount of percussion. Finally, layer 3 adds the melody line which only slightly changes the overall intensity of the music. Matching the progression of the gameplay information with the progression of change between the musical layers is key.

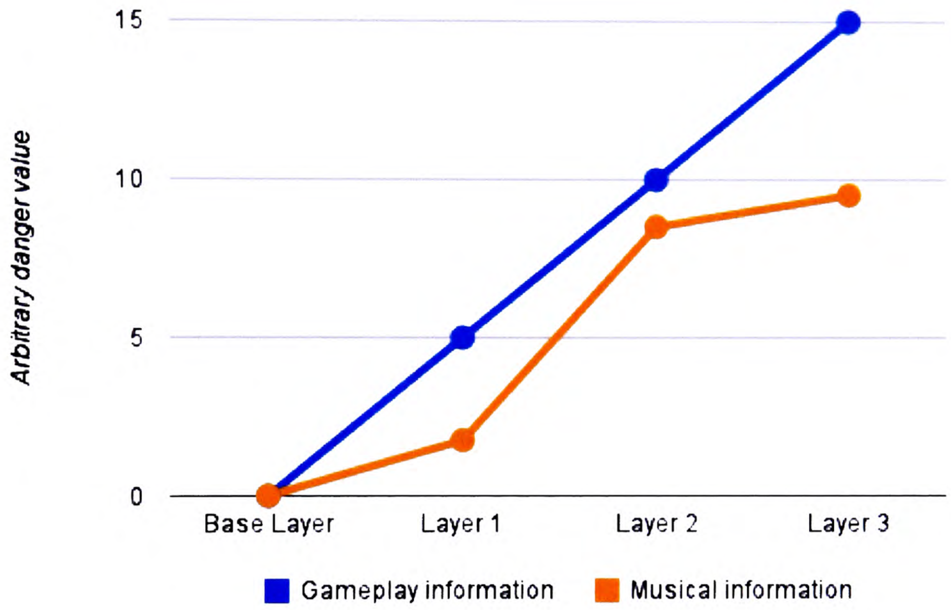


Figure 5.11 Mismatch between gameplay and musical information in Level 1.

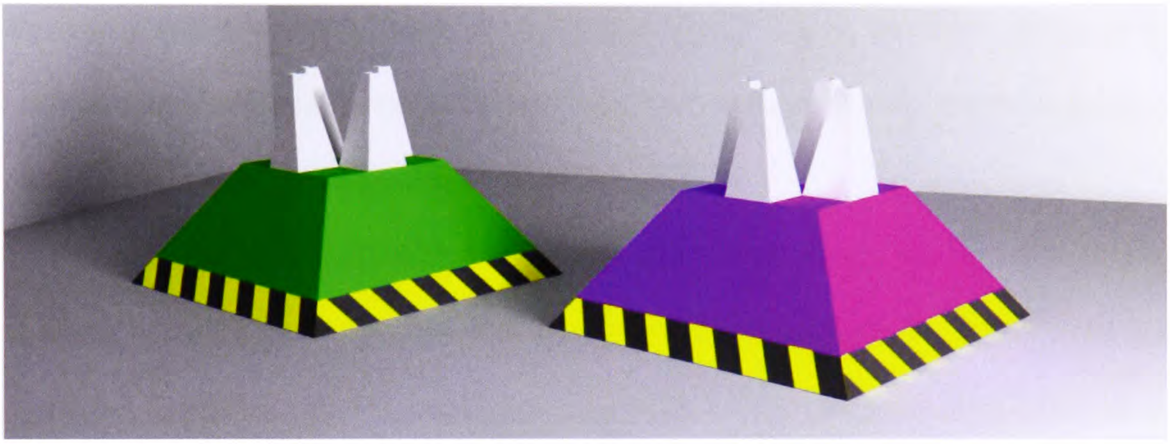
There are two possible ways of achieving this gameplay-to-musical-information match in the soundtrack, and they both centre on the idea of maximising the level of control the composer allows for in the music. The first method is to increase the musical representation of the information in each layer only in as small increments as possible (or necessary), maximising the control the designer has over the information in the soundtrack. This is possibly what makes the difference between the base layer and layer 1 work well. By only slightly changing the pizzicato part, the musical representation of the information – in this case, intensity – is subtly increased, but there is still plenty of room in the music, and on the scale of intensity, for further increase. Comparing this to what happens when layer 2 is added, the intensity is greatly and unnecessarily increased, completely out of proportion to layer 1. This could almost be thought of as layer 2 wasting potential intensity. In this specific case, the percussion could possibly have been spread over two layers. Not only would this have mapped the music to the information better, but also it would have increased the overall control of the soundtrack. While there is still room for further increase, layer 3, given its melodic nature, does not achieve this.

The second method of matching the progression of the gameplay information with the progression of change in the musical layers is essentially the opposite: where the musical representation of the information is controlled over as few layers as possible. Instead of controlling the layers in a binary manner, where they are essentially either on or off (discounting the slight fade to smooth transitions), the volume of each of these layers is used to manipulate the overall perception of the musical representation of the information. This is where layer 2 would work well: if, for example, it were played at low, medium and high volumes, these would create different levels of intensity, and the composition as a whole would be kept to fewer musical layers. However, it is unclear to what extent just controlling the volume of a layer will affect the perception of its intensity. This method would probably be highly dependent on the specifics of the music itself: it is likely that the impact of rhythmical parts changes to a larger extent through volume manipulation than harmonic elements, for example. Also, careful consideration would have to be given to the mapping or conversion of gameplay information to a musical representation and ultimately to a volume value (dbs).

This first level set out to investigate whether players can discern between individual layers of a soundtrack, and to some degree this appears to be the case. However, based on the results, as well as the player feedback, it became apparent that the music itself did not accurately reflect the information that it was attempting to convey. Therefore, two possible modifications to the compositional and implementational processes have been put forward and are explored further in Levels 2 and 3 (see Chapters 6 and 7).

6

Level 2: Vertical Layering with On/Off States



DESIGN

Designing Level 2 was an iterative process based on the findings of Level 1, which uncovered the fact that accurately mapping and matching the composition, and particularly the structure of the layers of music, to the information they are designed to represent is of primary importance in understanding the use of vertical layering as part of a game's information system. The goal of this second test, then, was to find a way to create a vertical layering soundtrack containing a more evenly spaced and consistent musical representation of the gameplay information using layer-spacing techniques as discussed below.

The Gameplay

Level 2 was primarily built on the foundations of the first level. Very little was actually removed from the first test: the majority of changes took the form of additional functionality and gameplay elements, all of which were explained in the Level 2 tutorial (see Appendix 2). As mentioned in Chapter 5, Level 1 was designed to be as simple as possible, but it seemed that more could be learned by adding further complexity, and it was clear from the first level that players would be capable of managing this additional depth of gameplay. Level 2 was therefore designed to introduce additional gameplay mechanics, giving the player more to do and creating a more realistic gameplay experience.

This goal was achieved by the introduction of the *micro-task*, which aimed to simulate, to a small extent, the concept of micro-management in RTS games (as discussed in Chapter 3). The micro-task consisted of two square platforms located in the player's base (see Figure 6.1). On each platform was a single, player-controlled unit referred to in the level as a *droid*. Beneath each droid was a corresponding spinning *micro-marker*, which randomly moved around on its platform every 20–30 seconds during the test. The player was given the task of controlling these two droids and attempting to keep them on their corresponding micro-markers. Not only did this micro-task add an extra layer of depth to the game, making it slightly more like a real RTS game, it also added an extra level of distraction for the player.

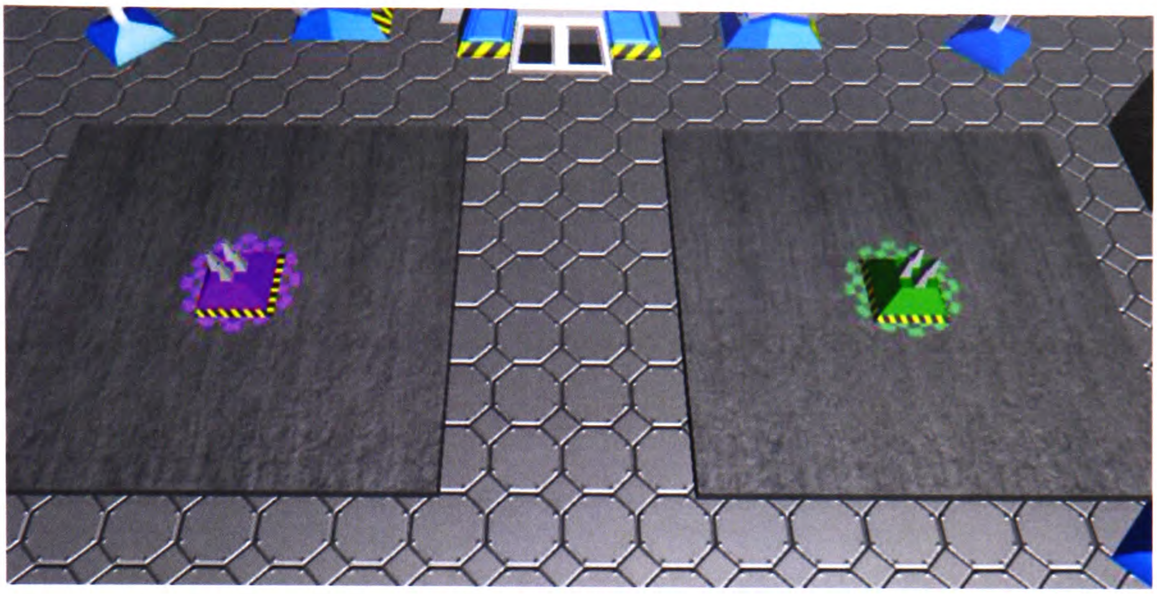


Figure 6.1 Micro-task in Level 2.

This gameplay interaction was monitored during the test and recorded along with the rest of the results. This data could be used to create a comparison between how well the player did in the micro-task and how well they did in the musical aspects of the test. The micro-task was measured by timing how long each droid remained on its respective marker. There was one timer that recorded how long the test lasted overall, from the end of the tutorial up until the final wave of enemies was destroyed, and another timer for each droid that only counted while the droid was within its micro-marker. The micro-task added a metric that could be used to represent how "well" the player was playing the game, which was something that was absent from the first level, and this would enable a comparison to be made with how well they performed in the task of predicting enemy wave sizes.

Further complexity was added to Level 2 in the form of enemy wave sizes. In Level 1, there were only three different brackets: 1–3, 4–6 and 7–9 enemies. In Level 2, two more brackets were added to make five different wave sizes: 1–2, 3–4, 5–6, 7–8 and 9–10 enemies. The players of Level 1 were able to cope with three different wave sizes relatively well; therefore, adding two more would increase the difficulty slightly and lessen the possibility of players accidentally guessing correctly. It is worth noting that rather than adding a 10–12 bracket and a 13–15 bracket, the size of bracket was changed (now two in each). This only added one more enemy to the largest wave size, meaning that the balance of the rest of the game was not greatly affected. For example, the health of enemy units in comparison to the damage of player units would not need to change to allow for any increase in enemy numbers.

With Level 1, there was a potential issue with having only three different wave sizes. During the tests, the players were *told* when a wave of enemies had spawned (they were not expected to predict this). Therefore, if a player could not tell the difference between the base layer and layer 1, then by simply being told that the wave has spawned, the player would probably realise that at least 1–3 enemies must have spawned, as this would create the least change in the soundtrack, even though they could not hear a difference in the music. It was hoped that including five different enemy wave sizes in Level 2 would make this less of a problem. It was thought that extra attention would have to be paid to the results for layer 1 to see how they compared with the other layers in order to determine whether this was an issue.

A final difference between Levels 1 and 2 was the wave-selection mechanic: how the test decided which wave size to spawn in each wave. Level 1 used a fully random system: as mentioned in Chapter 5, setting a wave-type order would inevitably have affected the results. Fully random spawning was thought acceptable for the first level which only had three different wave sizes. However, with Level 2 featuring five wave sizes, this would inevitably cause problems with the spread of wave selections. It was not desirable to increase the number of waves in the test as this would increase its overall length. As noted in Level 1 (see Chapter 5), there was already potentially some fatigue affecting the players' performance in the latter waves; therefore, it would not make sense to increase the overall length of the test. With a fully random selection method for ten waves, the best outcome would be for two of each wave size to be selected, which was not likely to happen for each individual player. While the distribution of wave sizes may well have balanced out over all of the players' results, having the potential for many players not to hear certain layers would spoil the ability to analyse individual player's results.

It was therefore decided that a *random without replacement* selection method would be used. In this system, once a wave size had been selected, it would be removed from the pool of wave sizes until all of the other wave sizes had been picked. This would guarantee that two of each wave size would be heard, while still fulfilling the desire not to set a wave spawning order common to all players.

The Compositional Process

The goal of the music in Level 2 was to create a more consistent mapping of music to information through the layers of the soundtrack, and, more specifically, to match the linear progression of the gameplay information it was representing: numbers of enemies.

A number of musical layer-spacing techniques were experimented with in the music of this level as ways of potentially spacing (from layer to layer) the musical changes in a vertical layering composition: that is, maximising the *perceived* difference, while minimising the actual *compositional* difference, both in order to space the distribution of the musical representation of information evenly and to make it as easy as possible for the listener to hear these differences.

These layer-spacing techniques are centred on the idea of three states: low, medium and high. First of all, the composer chooses a *primary spacing parameter*, which could be a musical parameter such as pitch, volume or note frequency (number of notes per bar). Starting with layer 1, then, the layer must be composed keeping the primary spacing parameter *low*; in layer 2, it should be *medium*; and then in layer 3, *high*.

Using the music composed for Level 2 as an example (see DVD item 07), the primary spacing parameter is pitch. Apart from the base layer (see Figure 6.2 and DVD item 08), which consists of two pad instruments that simply play drones, the remaining five layers use this low, medium and high spacing technique. In order to improve the evenness of layer spacing, unlike in Level 1, which featured multiple instruments per layer, the layers of Level 2 featured only one instrument per layer so that the total number of layers could be increased. Layer 1 simply contains a very low frequency kick drum (see Figure 6.3 and DVD item 09); layer 2 adds a mid-range frequency electronic snare drum-like percussion instrument (see Figure 6.4 and DVD item 10); and, finally, layer 3 adds a much higher, hi hat-like percussion instrument (see Figure 6.5 and DVD item 11). It is worth noting that when the parameters are referred to as low, medium and high, this is in a comparative sense only and not meant to be entirely prescriptive: the primary spacing parameter of one layer only needs to be *high* in comparison with the *low* layer, for example, or, equally, could be thought of as *higher* than the other two. The actual specifics are left to the discretion of the composer and will almost certainly be highly dependent on the various characteristics of each composition. It should be stressed that this suggested layer-spacing technique is put forward for the purposes of experimentation and not as a definitive set of rules.

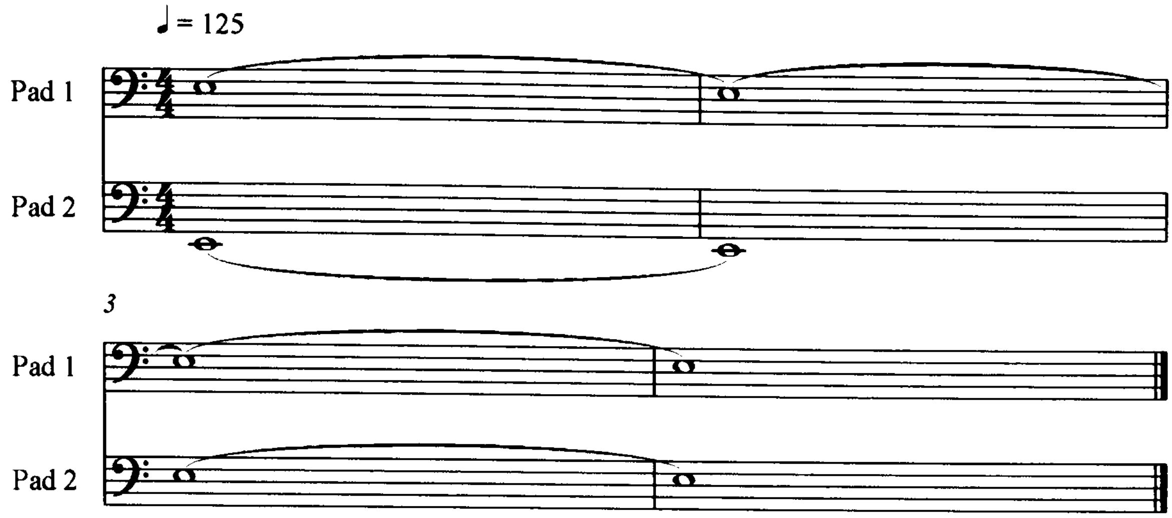


Figure 6.2 Level 2: extract from base layer.

Figure 6.3 Level 2: extract from layer 1.

Figure 6.4 Level 2: extract from layer 2.

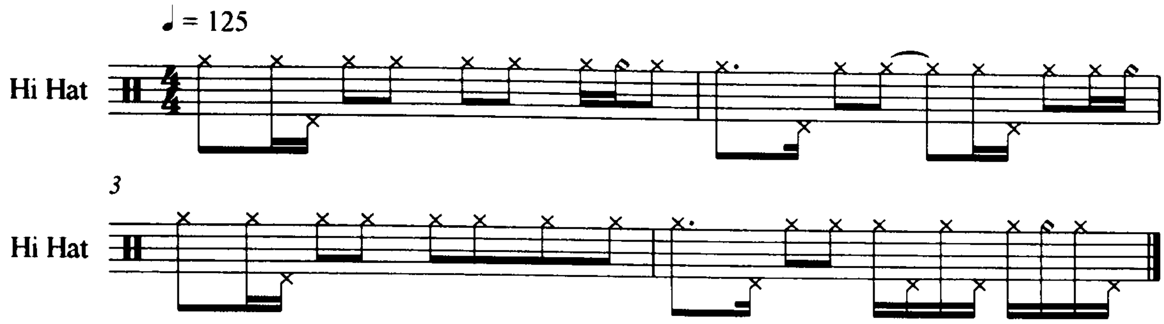


Figure 6.5 Level 2: extract from layer 3.

These three parts can then be thought of as one *spacing cycle*. For a composition that requires more than three layers, a second cycle must be introduced, only this time it is further manipulated by a *secondary spacing parameter*. This parameter helps distinguish the second cycle from the first, and can again be something like pitch, volume or note frequency, but must be different from the primary spacing parameter.

In the music for Level 2, instrumentation is used for the secondary spacing parameter. As can be seen in Figures 6.6 and 6.7, the instrumentation of layers 4 and 5 changes from the percussion of lower layers to synth instruments (see DVD items 12 and 13). The layers still follow the original low, medium and high pattern of the primary spacing parameter, but are distinguished from the first three layers because of the secondary spacing parameter. Note that in the music for Level 2, because there are only five layers (excluding the base layer), there is no *high* in the second spacing cycle. (They are referred to as Low and High Synths in the notation purely to differentiate them from each other and not to refer to the low, medium and high spacing.)

Figure 6.6 Level 2: extract from layer 4.

Figure 6.7 Level 2: extract from layer 5.

It should also be noted that, while the principle has been described here with each cycle containing essentially three parts (low, medium and high), it is possible that more than three parts may work: perhaps low, medium, high and very high, for example. But for the purpose of Level 2, the parts to each cycle were kept to a maximum of three.

The Music System

As mentioned in Chapter 4 on Methodology, at the time that Level 2 was being produced, Firelight Technologies, the developers of Fmod Studio, released a C-sharp wrapper for their product that would allow Unity users to integrate Fmod into their games. It was decided then that the original scratch-built music system from Level 1 would be replaced with Fmod which would mean that less of the audio functionality would have to be realised through scripting, meaning more complex behaviour could be created more easily. The structure of the original music system was designed to mimic very similar functionality to that provided by modern audio middleware. Therefore, making the switch to Fmod required very little in the way of changing workflow. Fmod Studio's multi-track event system can be easily used as a layer-based music system.

Initially, the plan was to control all of the musical layers via one *enemy count* parameter. This way, the original MetaObject could simply update an Fmod controller script whenever an enemy was spawned or killed and the single parameter of the music system would adjust. The volume automation curves, which would essentially turn the layers on and off, would be set up in a way shown in Figure 6.8.

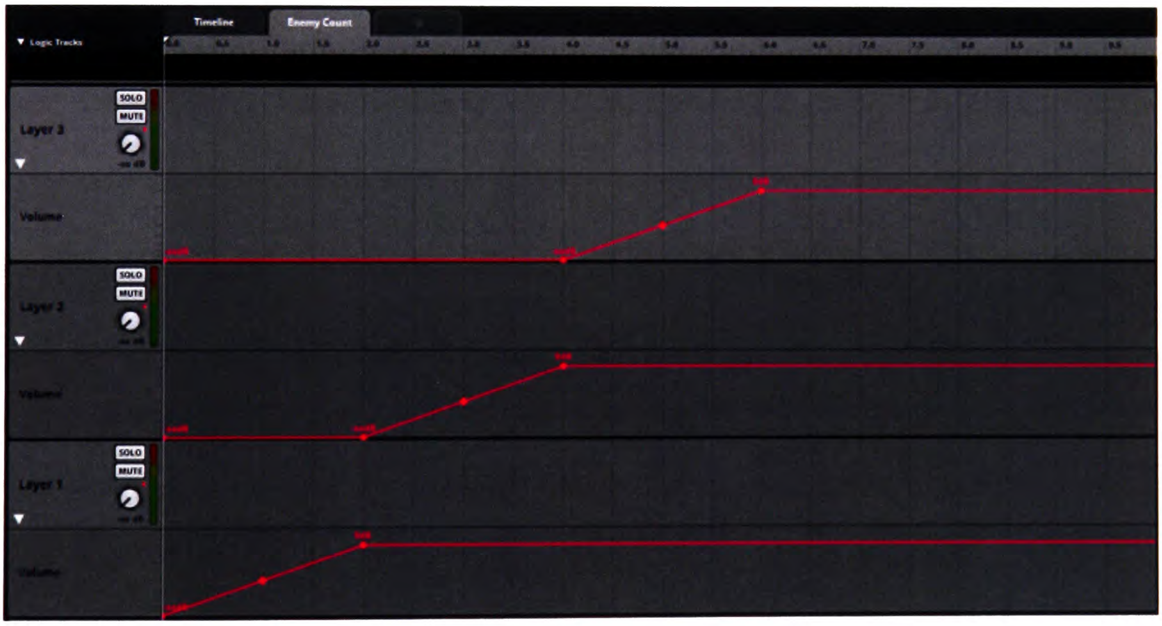


Figure 6.8 Original Fmod event set-up, showing volume automation of layers, in Level 2.

Setting up the music system in this way raised an interesting problem, however. By controlling the music with a single parameter, in this case enemy count, the time needed to adjust the music increases with each layer added, as the parameter has to track across each of the volume automation graphs that turn up the individual layers: for example, going from the base layer to layer 3 takes three times as long as going from the base layer to layer 1. Because of this, the only way to achieve the same effect as Level 1, where each layer adjusts its volume at the same time as the others, is to have one volume parameter for each layer and have the music controller script within the game contain the logic for which layers are turned on and when. This creates a slightly more complex five parameter system rather than simply one parameter, but was necessary to achieve the desired effect and had the added bonus that it slotted in more easily to the pre-existing code from Level 1.

Potential Findings

Overall, the difficulty of this second test level was increased. Players performed very well in the first level; they certainly responded better to the soundtrack than had been expected. Therefore, there was definitely scope to increase the complexity of the gameplay in this second test, and see how far the music system could be taken in this level in terms of the amount of information transfer, using layer-spacing techniques.

It was thought that correct prediction rates in this test would be much lower than in Level 1, but this test afforded the ability to look at how *close* players were with their predictions. Because Level 2 contained five different wave sizes, and thus predictions, it is possible to see whether the player was *one away* from predicting correctly: for example, if layer 3 played and the player predicted 2 or 4. This data might possibly reveal whether players were making educated predictions or just blind guesses.

What was hoped, though, was that, despite the potential lower correct prediction rates, overall the layers would be predicted correctly or incorrectly at a much more *even* rate. Whereas in Level 1, layer 1 had a much higher correct prediction rate than the other two layers, it was hoped that in this level the distribution of correct predictions would be much more evenly spread across all layers. This might go some way to show whether the soundtrack of Level 2, potentially because of the layer-spacing techniques, achieved a much more accurate mapping of the information it was trying to convey.

Finally, it would be interesting to see how the micro-task affected this test, as this was the first time the players would face a major distraction as they listened to the soundtrack and made predictions. Of particular interest was the relationship between how the player performed in the micro-task and how they performed in the music task, and whether players who did well in one also did well in the other or whether the two would be inversely related.

For a video compilation of extracts recorded from Level 2, see DVD item 14.

To play Level 2, see DVD Level 2: test executable folder.

RESULTS

In all, 15 players participated in this second test level. While a number of the participants from Level 1 dropped out, a few new players joined for Level 2. Overall, though, a similar number of players in total took part in the second test.

Table 6.1 gives an overview of each player's results. The data format remains the same as used for the Level 1 results: the first digit represents the wave size that spawned and the second digit represents what the player predicted. This time, as there were five different wave sizes, and likewise layers, the results ranged from 1 to 5, with a zero still indicating no prediction made. As with the discussion of Level 1 results, the following analysis will mainly talk in terms of *layers*, rather than enemy numbers or wave sizes, to keep the focus on the music. The final column in Table 6.1 displays each player's results from the micro-task. The percentage represents the amount of time the player kept both droids within their markers out of the total length of the test.

Table 6.1 Each player's results for Level 2

	Wave 1	Wave 2	Wave 3	Wave 4	Wave 5	Wave 6	Wave 7	Wave 8	Wave 9	Wave 10	Micro Task
Player A	11	44	55	23	34	44	23	11	55	33	87.4%
Player B	22	44	55	32	11	11	44	22	55	33	60.6%
Player C	53	32	21	43	11	21	11	44	55	32	73.2%
Player D	10	21	32	42	53	11	32	22	40	53	21.3%
Player E	43	21	11	33	55	33	22	55	11	44	95.0%
Player F	42	21	31	11	54	11	23	33	55	44	77.1%
Player G	11	22	44	55	33	22	55	32	11	44	73.7%
Player I	22	52	31	43	11	43	22	54	32	11	31.2%
Player J	11	53	22	40	32	44	22	55	11	33	64.1%
Player L	12	53	20	43	30	22	55	33	43	11	51.1%
Player M	43	54	20	32	11	42	21	33	11	55	81.9%
Player N	20	44	11	55	33	55	22	44	33	11	87.4%
Player Q	52	10	32	43	22	40	54	10	21	33	21.6%
Player R	11	32	43	20	55	54	21	11	42	32	41.2%
Player S	32	20	54	40	11	33	11	43	24	55	36.3%

Overall, 54% of player predictions were completely correct and, when including one-away predictions, 82% of waves were predicted correctly. With five possible predictions for each wave, a prediction percentage above 20% might suggest that players are not purely *guessing* but, similarly to what was surmised from Level 1, it is likely that players are using the soundtrack to aid their predictions.

How Even was the Layer Prediction Rate?

One of the primary goals of this level was to see whether the accuracy of player predictions could be made more even across all of the musical layers through compositional techniques. While overall accuracy was still deemed important, having the correct prediction rate of each layer comparatively even would potentially show that the musical representation of information was spread across the soundtrack more evenly than in the music of Level 1.

Figure 6.9 shows the percentage of correct predictions for each layer: 86.7% for layer 1, 43.3% for layer 2, also 43.3% for layer 3, 40.0% for layer 4 and 56.7% for layer 5. The results show that layer 1, in a similar way to the results of Level 1, had a much higher correct prediction rate than the other four layers, which were all comparatively similar. It was predicted that, because the player is informed when waves spawn, it might possibly be much easier to predict the first layer. This possibly contributed to the large disparity between the total correct predictions for layer 1 and the rest of the layers. Although it seems that the pattern of results is in fact the same as was seen in Level 1 – with the first layer being much higher in prediction accuracy than the other layers – this will be shown not to have been completely the case when *layer confusion* is discussed below.

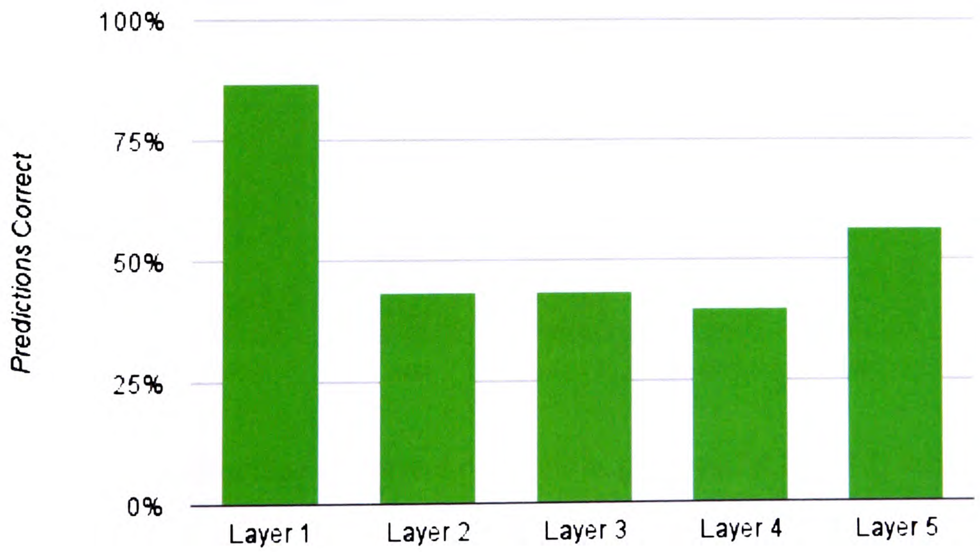


Figure 6.9 Percentage of correct predictions per layer in Level 2.

Figure 6.10 displays the percentage of correct predictions for each layer, but also includes one-away predictions (as described above) as correct. For example, if a layer 3 played, player predictions of 2, 3 or 4 would all be considered correct. As can be seen in Figure 6.10, prediction accuracy is far more even overall, and possibly shows that the players are making informed predictions and not just guesses. Interestingly, the other layers *catch up* with layer 1, and the amount of increase in layer 1 due to one-away predictions is small compared with the rest. This may be simply because layer 1 was already so accurately predicted that there was little room for improvement. Layer 5, although fairly close to layers 2, 3 and 4 in prediction accuracy, had a slightly higher correct prediction rate to start with, and likewise sees a slightly smaller level of improvement when including one-away predictions. Both of these facts could be because layers 1 and 5 only have one potential one-away.

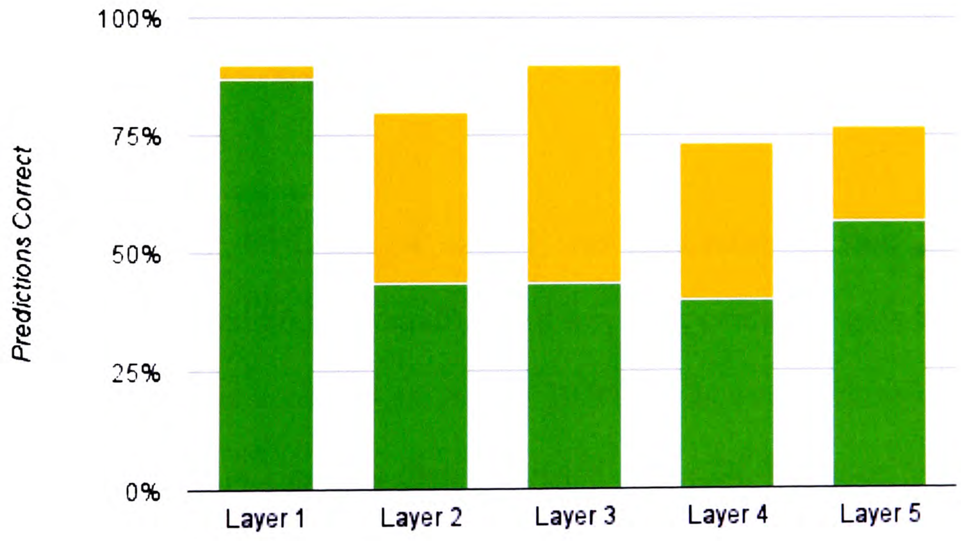


Figure 6.10 Percentage of correct predictions per layer in Level 2, including one-way predictions (yellow).

Confusion between Layers

Table 6.2 displays the confusion between musical layers. This shows the percentage of incorrect predictions for each possible combination of layers. Based on Figures 6.9 and 6.10, it could be argued that the layers were understood much more evenly in this level. However, looking specifically at what the various incorrect predictions were mistaken for might give further insight into any problems with the soundtrack. As with the results for Level 1, for the sake of simplicity the discussion here will refer to the layers as 1s, 2s, 3s, 4s and 5s.

Table 6.2 Confusion between layers in Level 2

Layer Predicted \ Layer Played	1	2	3	4	5
1		26.7%	6.7%	0.0%	0.0%
2	3.3%		43.3%	13.3%	6.7%
3	0.0%	10.0%		33.3%	16.7%
4	0.0%	3.3%	3.3%		20.0%
5	0.0%	0.0%	0.0%	0.0%	

First, the confusion of layer 1 was unsurprisingly low, given that it was correctly predicted 86.7% overall. There was more confusion with regards to layer 2: 26.7% of 2s were confused as 1s, and 10.0% of 2s were confused as 3s. Likewise, layer 3 contains a similarly high confusion rate with 43.3% of 3s confused as 2s. Again, layer 4 contained a similar confusion rate with 33.3% of 4s confused as 3s. Finally, 20.0% of layer 5 were confused with layer 4, but there was also some confusion between layers 3 and 2 (see Table 6.2).

Overall, the highest amounts of confusion were: 2s as 1s, 3s as 2s, 4s as 3s, and 5s as 4s. What is interesting, though, is that, unlike the results for Level 1, where layers 1 and 2 as well as 2 and 3 were roughly equally confused in both directions, this does not appear to be the case in this level. While 26.7% of 2s were confused as 1s, only 3.3% of 1s were confused as 2s. Similarly, 43.3% of 3s were confused as 2s, while only 10% of 2s were confused as 3s. Continuing the trend, while 33.3% of 4s were confused as 3s, only 3.3% of 3s were confused as 4s. Finally, 20.0% of 5s were confused as 4s, while no 4s were confused as 5s.

There is an obvious pattern here that layers are more often than not being mistaken for the layer below them. This could indicate that, overall, the players thought the soundtrack was not quite intense enough to represent the numbers of attacking enemies that it was attempting to. However, because the layer confusion is only *one down*, it might appear that the musical representation of information in the soundtrack was close to being appropriate for the number of attacking enemies or at least close enough for the players to interpret it this way.

Another way to view this is to look at the overall spread of predictions, regardless of whether they were correct or not. Table 6.3 displays the total number of predictions for each layer, including both correct and incorrect predictions. Starting with layer 1, chosen a total of 36 times amongst the participants, each subsequent layer gradually decreases in the number of total predictions, with layer 5 being chosen less than half as often as layer 1. This pattern reinforces the possible idea that, while each layer was more evenly predicted correctly in this level, overall the players felt that the musical representation of information, in this case intensity, was not enough to match the numbers of enemies in each particular wave.

Table 6.3 Total number of predictions for each layer in Level 2

Layer number	1	2	3	4	5
Number of times chosen	36	33	30	20	17

Player Performance over Time

Gauging whether or not players were improving over the course of the level was another area of interest. While there was certainly some evidence in the results for Level 1 that might suggest players improved their prediction performance over the course of the test, if the same pattern was present in this level, this would reinforce the notion that players can improve their understanding of the soundtrack during play.

For each wave, all of the correct predictions for each player have been combined, regardless of layer/wave size, and are displayed in Figure 6.11. As in Figure 6.10, this figure also shows the total correct predictions per wave including one-away predictions as correct. As can be seen, in both cases, on average, the trend is towards higher numbers of correct predictions the further into the test that the player has reached. Interestingly, similar to Level 1, the peak of correct predictions is around waves 7 and 8, with a tail-off afterwards; however, in this level, the number of correct predictions slightly increases on wave 10.

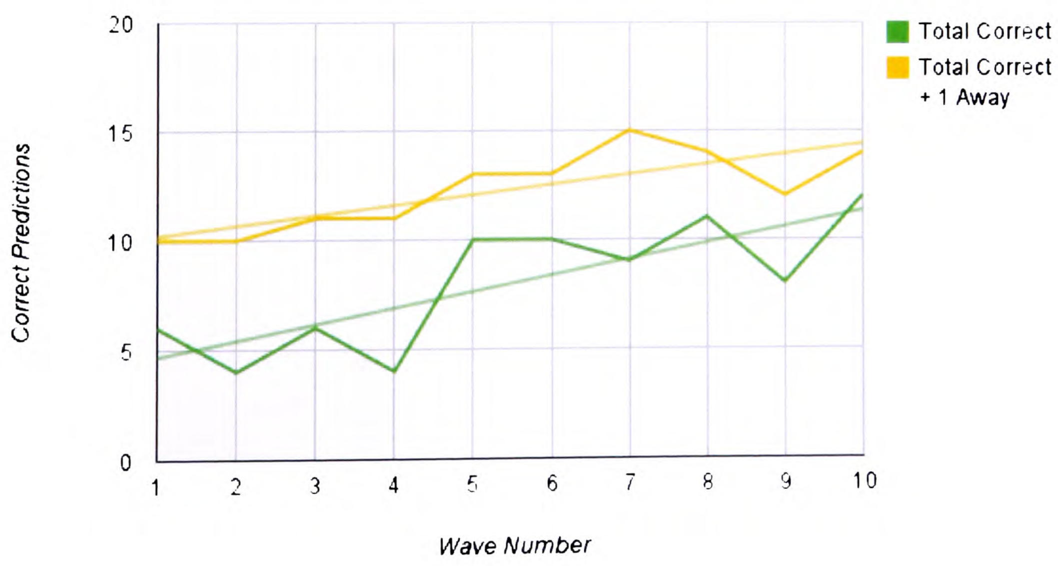


Figure 6.11 Player performance over time in Level 2.

Overall, it is believed, then, that this reinforces the notion that players are able to learn the soundtrack over the course of the test. Even if they did not manage to memorise each layer fully during the tutorial, the more waves they face and layers they hear, the more comparisons between them they are able to make, thus improving the overall accuracy of their predictions.

Comparison between Prediction Performance and Micro-task

Included in this level was the micro-task, which not only acted as a potential distraction for the player and created a more game-like experience in this test, but also created a metric which could be used to compare the players' prediction results with how well they performed in the micro-task – how good they were at *playing* the game.

Figure 6.12 displays this comparison, with each dot representing a player. Each point on the x axis corresponds with the player's total number of correct predictions, and its position on the y axis corresponds with their performance in the micro-task. The percentage here represents the average amount of time that the two droids were kept within the micro-markers out of the total length of the test – each test lasting a marginally different length of time.

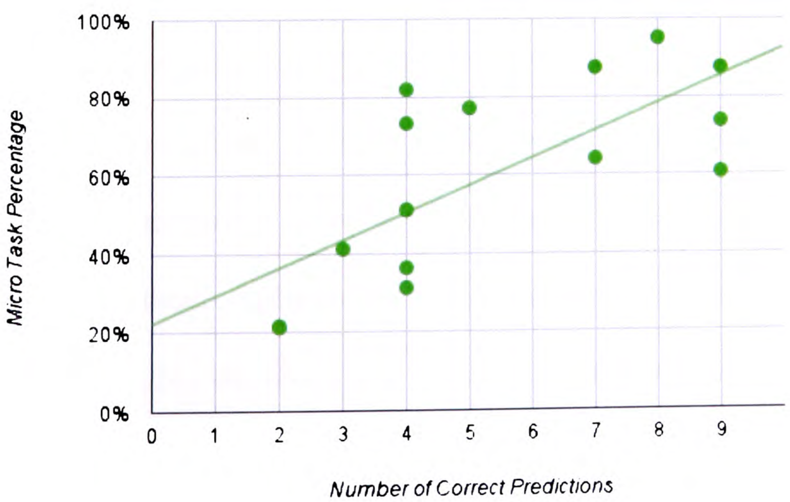


Figure 6.12 Comparison between prediction performance and micro-task in Level 2.

As noted in the Design section above, these two aspects of the test – the ability to understand the soundtrack and perform the micro-task – potentially had two relationships. The first was that the player's performance in one task would be similarly good or bad as their performance in the other task, relating to how proficient they might be at playing real games. The second possibility was that players would only perform well in one aspect of the game, but not the other. Either players would focus on the predictions and not be able to attend to the micro-task, or they would be too distracted controlling the droids that they would not be able to concentrate on listening to the soundtrack or might even forget to make predictions.

Figure 6.12, however, shows a clear relationship between how well they understood the soundtrack and how well they performed in the micro-task. Therefore, the former point, that players will either do well in both tasks or neither seems to be the case. This would appear to be a positive finding as it demonstrates that using the soundtrack as a means of communicating with the player does not necessarily have to be at the expense of the gameplay experience. However, this does not show whether the players would have performed *better* in either task had they not had to do the other one.

Is Player Performance Affected by which Layer is Heard in Wave 1?

As noted in earlier sections, both Level 1 and Level 2 featured random wave spawning mechanics. Level 1 featured a fully random system, while Level 2 used a random without replacement selection method. This created a large amount of variation within the levels and eliminated the potential negative influence of setting a particular sequence of wave spawns. It did, however, create the possible drawback that all players had a slightly different experience of the level and, more importantly, the soundtrack. The main difference is the order in which the players hear the layers. Although in this level players heard all five layers before they heard a repeat, which layer was heard first, second, third, fourth and fifth was likely to have had an effect on their overall perception and understanding of the soundtrack.

It was therefore a concern that a player who, for example, hears layer 1 first may well form an overall different picture of the soundtrack from a player who hears layer 5 first. It was therefore necessary to look at each player's performance in the prediction task, and compare this with which layer they heard in wave 1. Figure 6.13 displays this information: each player is represented by a point on the graph, with its position on the x axis showing their total number of correct predictions and its position on the y axis showing which layer they heard in wave 1.

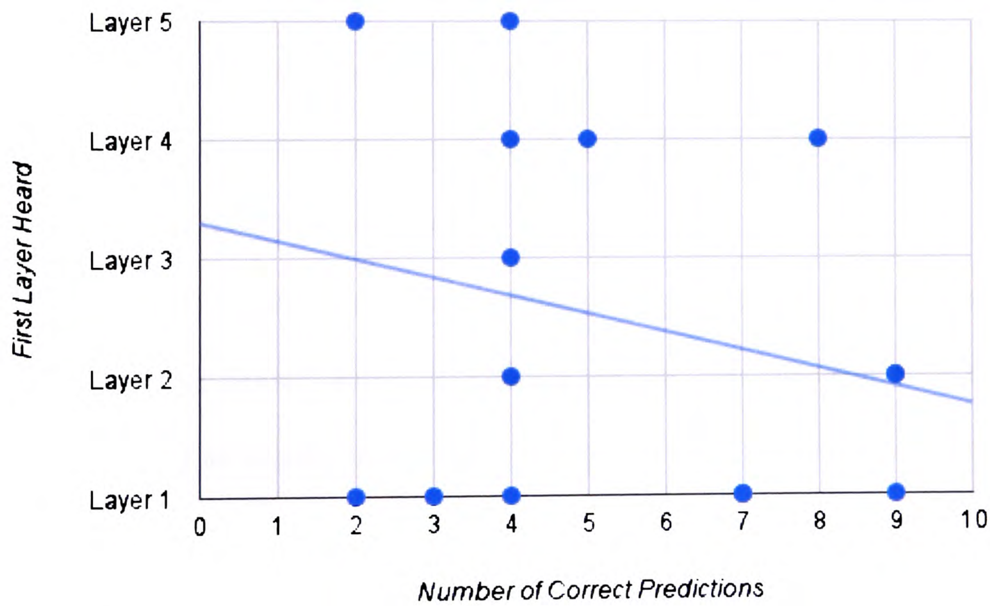


Figure 6.13 Comparison between first wave and overall performance in Level 2.

The data shows a slight trend towards players who heard a numerically low layer first performing better overall in the prediction task. It is possible that hearing a numerically high layer (and therefore in addition all the layers below it) in an early wave could be slightly overwhelming and more difficult to process for players, thus resulting in poorer performances. The overall spread of results and the limited number of players who took part would suggest that more data is needed before any conclusions about this aspect of the soundtrack and the testing method can be made. However, it could be argued that any potential effect this might have had on the test was mitigated by the fact that all the players heard the music for the first time in the tutorial in exactly the same manner.

FEEDBACK

Comparisons between Layers

As discussed in Chapter 5 on Level 1, players have the opportunity and ability to make comparisons between layers and use this as a way of calibrating their understanding of the soundtrack during play. The further the player progresses through the test and the more layers they hear, the more comparisons they can make within the layers of the soundtrack.

This point came up in discussion with Player E, who realised that he had been predicting "one below" before he heard a layer 1. This seems to back up the pattern viewed across all results that players were on average predicting one lower than the actual layer that they were hearing (see Table 6.2). In Player E's case, Table 6.1 shows how he realised this and readjusted his understanding of the layers.

After working out the way in which the results were formatted on the results panel at the end of the test, Player E was able to use these numbers to describe his thought process during the test: "The first one I was like, one below, then one below, then I realised what the first one [layer 1] was and I worked it out from there really."

What he is describing here relates to what was discussed with regard to Table 6.2. A 4 spawned in wave 1 and he chose 3; then in wave 2, a 2 spawned and he chose 1. So far, he has consistently predicted one below. But then in wave 3, a 1 spawned, and he realised he was wrong with his first two predictions and correctly chose layer 1.

Because of the uncertainty of his first two predictions, Player E was still thinking in terms of calibrating his predictions based on comparisons between layers as the test progressed when he said: "There was only one 4 apart from the beginning. I was thinking, have I had a 4? Have I got them all wrong because a 4 hasn't come up? And then 4 was at the end and I was, like, yeah I'm pretty sure this is 4!"

Describing the Layers

Players were asked to describe, retrospectively, what each individual layer had sounded like, using any descriptive language, whether musical or otherwise. It was believed that understanding how the player might describe a layer would give insight into how the layers were perceived. For layer 1, Player G used the description "a low, thumping sound" which accurately describes the bass drum of layer 1. He recalled layer 2 as "some sort of drum", which again matched the snare drum that comprised the layer. Player G struggled to remember layer 3. Given that layer 1 is heard in isolation and layer 2 only has layer 1 to distract from it, it might be that these are the easiest to hear and recall. However, Player G could recall layer 4, which he described as being "wobbly" and having "movement". While these are fairly abstract ways of describing the low electronic instrument's sound, in fact the instrument is processed with an automated panning and delay effect which arguably creates this "wobbly" feeling to the sound. Finally, Player G described layer 5 as being "synthy" and "electronic", an accurate description for the lead synthesiser.

The instrument contained in layer 3 is an electronic hi-hat, and this is possibly partly why it was the hardest for Player G to remember and describe. It is arguably quite unlike its acoustic counterpart as this particular instrument had considerable additional effects processing applied to it. Potentially, if a player cannot ascribe a word to a particular instrument, it may be harder for them to memorise and recall it. Players are likely to know the sound of a drum and perhaps can even describe a synthesiser, but layer 3 was possibly not so easy to identify. Player G got both of his layer 2 predictions correct, but did choose a "3 as a 2" for his only incorrect response; therefore, it looks as if there may have been some confusion or uncertainty around this layer.

Emphasis on the Micro-task

As noted earlier, it was a concern that players might not understand the micro-task or think that it was not a necessary part of the test. There is no feedback given to the player on whether they are succeeding or failing in the micro-task other than simply viewing whether the droid is in its marker or not. The tutorial states that players should endeavour to keep the droids within their respective markers, although it does not inform the player as to the purpose of the task and whether it has any bearing on any other aspect of the test.

Player E was curious as to the purpose of the micro-task when in his feedback he stated: "I wondered if you were recording it or if I was just wasting my time fiddling around following it! Or it was trying to distract my brain from what I was trying to hear with how many things were coming." Player E did, however, play along despite his suspicions that the micro-task might not matter. Player G also reported taking the micro-task seriously, although he claimed that he prioritised the predictions over the micro-task. This was the attitude that it was hoped most of the players would adopt, where they understood the micro-task was an important part of the test but the focus was on predicting the enemy waves.

Player N found the micro-task particularly distracting and obviously put a lot of focus on it, explaining that he missed the first prediction because of the droids. The fact that Player N missed the first wave in a sense justifies having the micro-task as it shows that this interaction did work as a distraction to the rest of the test. Player N also added that he thought there should be an "annoying noise" that could play whenever the droids were out of their markers – in a sense sonifying the micro-task. While this was beyond the scope of this test, it is interesting to see a player thinking about the use of sound as information in a game.

Approaches to Understanding the Soundtrack

Much of the player feedback centred on the ways in which they attempted to understand the music in this level. Interestingly, some players who took both tests made comparisons with how they approached this level compared with Level 1. Player E noted that he paid more attention to the layers of the music this time, having relied on his gut instinct more in the first test. It seems that being a player who had participated in both levels may have had an effect on his experience as he noted trying a different approach in Level 2.

He noticed that each layer added a new instrument to the music, and so he was able to count how many instruments he could hear and did not feel as though having to memorise the music was as important as counting the instruments. It is possible that this was in part due to the way the tutorial presented the music. Starting from the base layer, it introduced each subsequent layer, allowing the player to hear the changes clearly. Player E in this case noticed that each layer was adding a new instrument. He clarified, though, that this method of counting the layers constituted roughly half of his method of understanding the soundtrack; the other half he referred to as listening to "what it feels like", as he put it.

Player E was one of the players who achieved a high correct prediction rate (8/10) and therefore it is clear that his way of understanding the soundtrack worked. However, not all players managed to achieve such a high prediction rate, and it is likely that others did not realise this method of counting the number of instruments at any given time, revealing how many layers were being played.

The Advantage of Musical Training

Player N also was able to distinguish between each of the layers, but for him this was for a different reason. As a musician, Player N felt that he was at an "unfair advantage", and was able to dissect the music very easily. Having musical training does not usually have much of an impact on games, but this could be the case with this use of music. Player N was also asked about the impact of the music: whether the feeling or emotional effect helped him with his predictions. But, as he reported, he felt that the music was so easy for him to deconstruct that he did not need to use any of the feeling or impact of the music to help understand its information. He could simply remember and recognise each individual layer. Perhaps musicians cannot help but listen to music as a musician, and will potentially not hear and interpret the music in the same way as a non-musician. This is also likely to be the case when listening to game music.

CONCLUSION

Level 2 aimed to add further complexity to the foundations laid in Level 1. The number of wave sizes increased in Level 2, raising the difficulty of the level and lessening the possibility that players would be able to predict wave sizes correctly by chance. The micro-task was introduced to create a more game-like experience and to act as a distraction for the player, the results of which were recorded and compared with the prediction results.

The music of Level 2 was designed in such a way that the musical representation of gameplay information would be more evenly spaced across the musical layers than it had been in Level 1. The aim was to use the layer-spacing techniques discussed above to achieve this. As mentioned in Chapter 5, there are two possible ways of composing music for vertical layering: many simple layers that are turned on and off, or fewer, more complex layers controlled by their volumes. Level 2 aimed to explore the former.

It was initially thought that overall correct prediction rates would be lower in Level 2 given the increased difficulty of the level and complexity of the soundtrack. However, it was thought that, despite this, there might be a more even correct prediction rate across the layers. Although it was not the goal of the level to prove that the layer-spacing techniques were responsible for any improved results, as this would be too ambitious for the scope of this research, it would at least highlight some of the important aspects of vertical layering with regards to mapping music to information.

It was thought that, due to the addition of more wave sizes and therefore more layers of music, Level 2 could potentially give better insight into how players interpret the music. Including more layers meant that players had less chance purely to *guess* correctly and also created the opportunity to analyse the results in terms of how *close* the players were to predicting correctly. Finally, it was thought that Level 2 would reveal any link between how well players did in the micro-task and how well they did in the wave-prediction task.

What was Discovered?

In terms of the overall goal of layer evenness, the results do initially appear to be relatively even with the exception of layer 1, which may have had a disproportionately high correct prediction rate due to the test informing the player when waves had spawned. Although the results could initially be considered similar to Level 1, the analysis of layer confusion – where incorrect layer predictions were consistently confused with the layer below (but not both ways as was seen in Level 1) – potentially points to the fact that the soundtrack has improved its information transfer and that the high percentage of correct layer 1 predictions was caused by the announcement of wave spawns.

The notion that players improve in their predictions the more they listen to a soundtrack, which was evident in the results for Level 1, has been reinforced by the results of this level. Level 2 also afforded the opportunity to look at *one-away* predictions. This arguably showed that players were making informed decisions based on either their memorisation of the individual layers or their comparative interpretation of the soundtrack as a whole, and not pure guesses, even if they did not predict 100% accurately.

The comparison between the micro-task and the wave predictions showed that there seems to be a link between proficiency in multi-tasking and ability to interpret the soundtrack. Roughly speaking, players who did well in one also did well in the other. Which layer a player heard first during the level might have had an effect on how well the player performed overall in the wave-prediction task. The results for this are by no means clear and further research would be needed to determine this: the topic could potentially be an interesting area of research in itself. However, this may well be an important consideration when using vertical layering music in general.

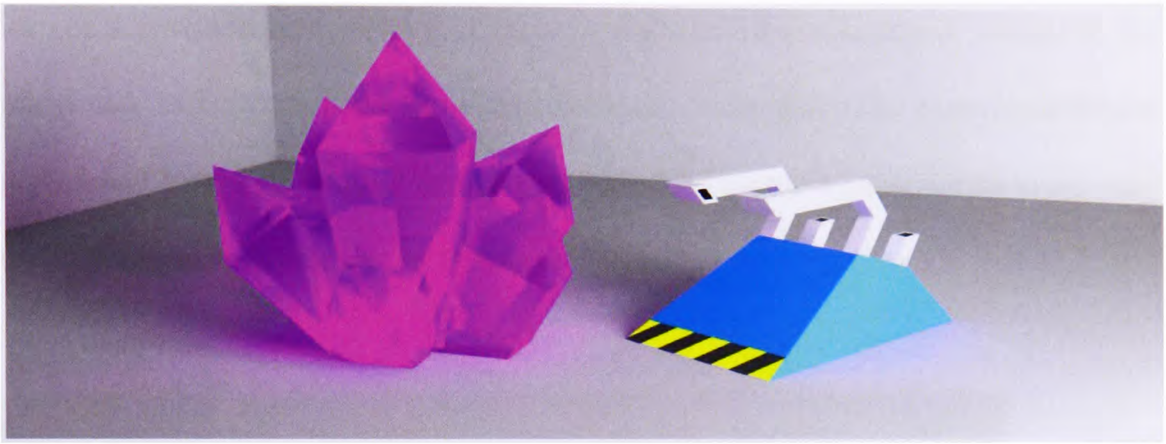
Discussions with players reinforced the notion that the musical representation of information – in this case, intensity and danger – was not strong enough in the soundtrack. This was evidenced by the pattern of responses frequently being *one down* from the correct prediction. Feedback also revealed that players constantly reassess their perception of the music based on comparisons between the music of the current and previous waves.

The issue of prior musical training giving certain players an advantage has been raised. While this does have certain implications for comparing individual results, more importantly, in terms of using music as information, this issue might only be relevant in contexts where players were competing directly with other players. These multi-player considerations have been discussed in Chapter 3, and far from being a problem with the use of music as information, should be treated as a game design decision.

Finally, feedback revealed a potential weakness in the way in which the music of Level 2 was structured. When each layer of the soundtrack comprises only one instrument, players can potentially count how many instruments they can hear and use this information to equate to the wave size or layer they are predicting. However, this is not necessarily a negative aspect as it demonstrates players' ability to deconstruct the music, which is the primary goal of this style and use of music. But perhaps simply counting how many instruments can be heard is too simple and detracts from the overall effect of the music. However, based on the results, it would appear that not many players realised this about the music, although it is still a relevant aspect to be considered when composing a vertical layering soundtrack.

7

Level 3: Multi-dimensional Information



DESIGN

Following the findings of Level 1 (see Chapter 5), it was decided that two potential ways of composing and implementing vertical layering soundtracks would be explored in further test levels. Level 2 explored the first method: composing the soundtrack in many simple layers that are controlled by setting them either on or off. This final test, Level 3, explored the second method: where the music is arranged in fewer layers but the volume of these layers is used to convey information. The use of *motifs* to convey information (as discussed in Chapter 3 in connection with *Left 4 Dead 2*) was introduced into the informational layers of Level 3 in order to convey information regarding enemy types. Therefore, individual layers would represent specific enemy *types*, while their volume would represent the *number* of such enemies.

The Gameplay

The overall premise of test Level 3 remained similar to the first two levels. The player was still asked to make predictions of enemies attacking their base, while they defended it and performed other tasks. As well as the micro-task from Level 2, Level 3 also saw the addition of the *macro-task* (see Figure 7.1) which aimed to mimic the idea of *macro-management* in RTS games, as discussed in Chapter 3. Similar to the micro-task, this aspect of the test aimed to create a more game-like experience for the player, add further distraction and give an additional metric against which to measure the player's wave-prediction performance.

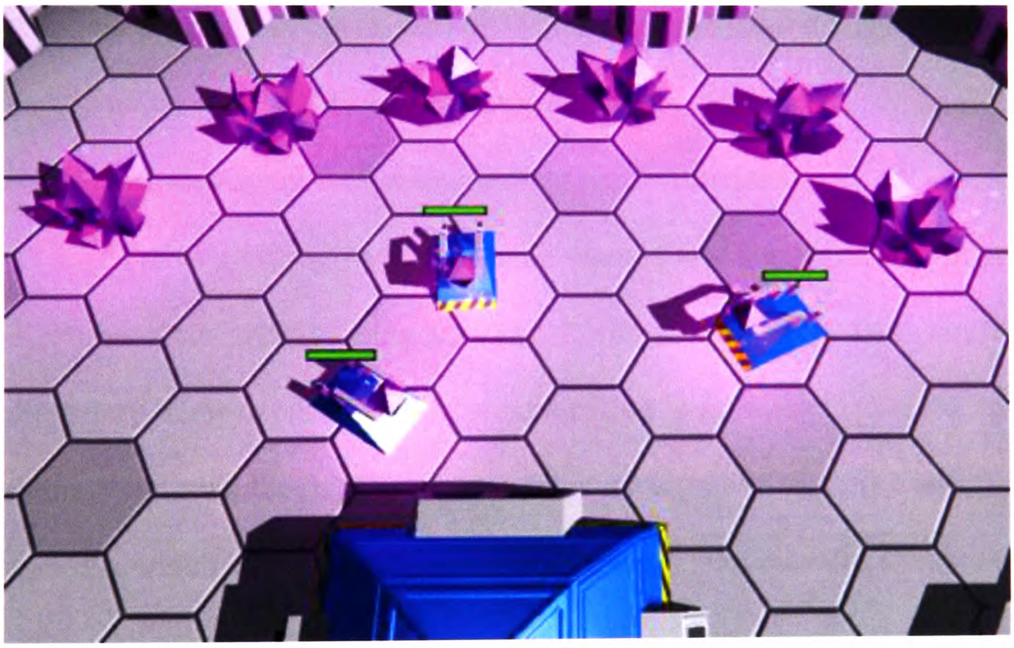


Figure 7.1 Macro-task of Level 3.

As can be found in many RTS games, the macro-task centred on the player's *worker* units collecting resources and returning them to the player's main base structure. Each time a predetermined amount of resources had been collected, the player had to spend them by pressing a button in the bottom right-hand corner of the screen. Spending these resources either repaired one of the player's defensive turrets, if any had been destroyed, or reinforced their army with a new unit. Collecting resources and creating units is a major part of RTS games (see Chapter 3) and something that had not been included in the test levels up to this point. It was therefore believed that this would make the test feel much more like a real game for the player.

The player's proficiency in performing the macro-task was measured by recording how long it took for the player to spend the resources. When the target number of resources had been collected, this amount was frozen and a timer was started that counted until the player pressed the "spend" button. The time was then recorded with the rest of the results in a similar way to the micro-task. The macro-task relied on the fact that the player noticed they had collected enough resources to spend, and therefore there would be a variable amount of results gathered from players based on whether they spent their resources more or less frequently.

The way enemy wave spawning worked was significantly changed for Level 3. Rather than simply being concerned with enemy wave size, this test added a secondary dimension to the information that was being conveyed to the player through the soundtrack. The enemies now consisted of three different types: *yellow enemies*, which were small in size, fast, but weak; *red enemies*, which were medium-sized with average speed and strength; and, finally, *brown enemies*, which were large, slow, but very tough (see Figure 7.2). Each of these different enemy types could attack in either a small group or a large group in each wave, a combination of which was selected at random (without replacement) by the wave spawner. For example, wave 1 might consist of a small group of yellows, a large group of reds and a small group of browns. It was also possible for a certain enemy type *not* to be selected for the wave by the game: wave 2 might consist of a large group of yellows, a small group of reds and no browns at all. This created many different combinations of overall wave types for the player to predict.

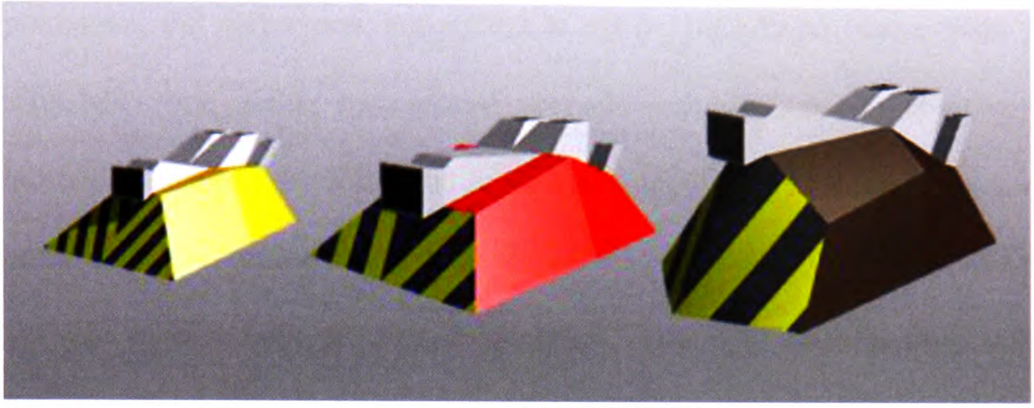


Figure 7.2 The three enemy types of Level 3.

Two exceptions to these wave combinations were made, however. It was decided not to allow for only one enemy type to be present in a wave given that this test wanted to see how well the player could interpret a complex soundtrack. It was also decided that three large groups would not be allowed to spawn. This was in part due to game balance – so the player was not completely overrun should this combination spawn in an early wave – but was also due to musical considerations, in particular the perceived difference in loudness created by the layers representing a wave of two small groups compared with a wave of three large groups. This exemplifies how, when gameplay and music interrelate, the designer needs to consider how one will affect the other. In this example, both the gameplay and the music dictated what type of enemy waves could spawn. The player, however, did not know that these exceptions were being made: they were simply asked to predict the numbers of each enemy type present in the attacking wave.

The Testing Mechanics

To accommodate the increased complexity of the predictions, the test response buttons were also reworked. Rather than simply pressing a single button to make a prediction, the player now had access to six *toggle buttons*. These were grouped in three pairs (one pair for each enemy type) and each pair consisted of a small and large button, referencing the small and large wave sizes. The player therefore predicted the composition of the wave they were currently facing by selecting a combination of the six buttons and then pressing the "submit" button to confirm their selection (see Figure 7.3).

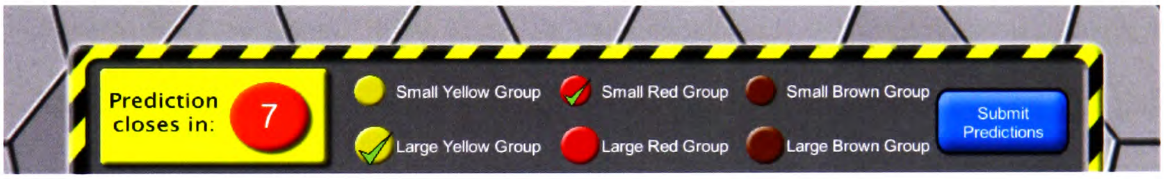


Figure 7.3 Prediction toggle buttons for Level 3.

The results were stored using a three-digit number that was assigned at the point the player pressed the submit button. The first digit (the 100s) was used to denote the yellow enemies, the second digit (the 10s) the red enemies and the third digit (the units) the brown enemies. Starting from zero, a certain value was added to the total depending on how the toggle buttons were set when the player pressed the submit button. For example, for the yellow buttons, if small was selected, 100 was added to the total. If large was selected, 200 was added to the total, and if neither was selected, zero would be added to the total. The same pattern was used for the red toggles, only with 10, 20 and zero being added, depending on toggle state; and 1, 2 or zero for the brown toggles. An identical process was used to store the results for the combination of enemies that actually spawned in the wave and these two three-digit numbers created a simple way of recording and retrieving the players' results.

An early concern was that players might forget to submit their results before the time ran out. This was understandable given that previous tests required only one button press, and this test required at least three, including the submit button, to be pressed. Initially, it had been decided not to *auto-accept* players' results if they did not press the submit button, as this might give inaccurate responses. Specifically, if the player was halfway through selecting from the toggle buttons, and assuming that players would make their selections from left to right, the brown enemies would be the last to be chosen, making them the most susceptible to problems from auto-accept. However, because the potential of losing valuable results outweighed the negatives, a solution was devised that would accept the currently selected predictions, but that would also show that an auto-accept had taken place. This way, more scrutiny could be applied to those results that were auto-accepted, without losing them entirely. This was achieved by adding another digit to the results. If an auto-accept of the predictions happened, 9000 would be added to the results. The arbitrary "nine" at the start of the result digits would therefore denote an auto-accept. Despite this method, it was still possible that auto-accepted results would have to be discounted, but at least this solution allowed for an informed decision to be made.

The total number of waves that the player would face was also adjusted for Level 3. Because there were more enemy units present in the waves, each wave would take longer to arrive and be destroyed. Therefore, in order to keep the overall test length similar to the previous tests, the total number of waves was reduced from ten to six. This was deemed to be a necessary sacrifice in order not to overburden the player with the length of the test.

The tutorial at the start of the test (see Appendix 3) also featured a slightly different method of playing the example music. In previous tests, the player was introduced to the music by hearing one prerendered file, which was arranged in such a way that each layer was vocally introduced and then faded in to the rest of the music. This time, the player was presented with four buttons (see Figure 7.4), which, when pressed, played a short extract of the musical layer that represented the particular type of enemy pictured on the button. The fourth button, when pressed, played an extract that included all of the layers, including the other, non-motivic information layers. This essentially allowed the player to hear the music that represented the different enemies in isolation, as well as hearing one possible combination of the layers as an example of how the entire soundtrack might sound. Pressing the buttons allowed the player the freedom to listen as many times as they wanted in order to understand the music.

Figure 7.4 Level 3 tutorial music example buttons.

The final major change seen in Level 3 was the visual overhaul. All of the simple cubes that made up the level geometry were removed and replaced with custom-made models to improve the aesthetic and visual style of the test (see Figure 7.5). The goal was to improve the overall look of the level and create a less prototype-like and more convincing game-like appearance. The importance of the visual appeal of the level should not be underestimated in its effect on audio perception. Even though this research is audio focused, the players would no doubt be affected by the visual aspects of the test.

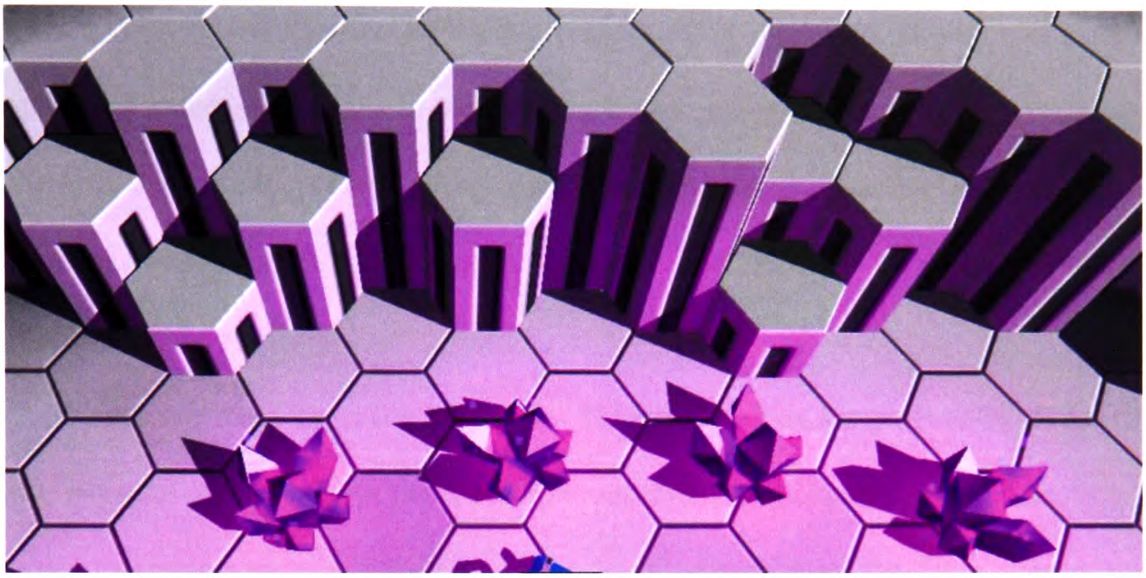


Figure 7.5 New models for Level 3.

The Compositional Process

The music of Level 3 consisted of two types of layer: *informational layers* and *game state layers* (see DVD item 15 for an example of the music in Level 3). The three *informational layers* corresponded with the yellow, red and brown enemies, and only played when those particular enemies were attacking. These were thematic parts that were inspired by the low, medium and high aspect of the layer-spacing techniques discussed in Chapter 6. These layers were able to play in any combination to work with the random method of enemy spawning. The presence of these layers conveyed motivic information, while the volume of a particular layer conveyed in what size of group that enemy type was attacking: high volume denoted a large group; low volume a small group. This was a musical representation of information, similar to that of the previous two levels, where a greater sound of danger and intensity meant a greater number of enemies: higher volume equalled higher numbers of enemies.

The various characteristics of the enemy types were also used to inspire the composition of the motivic information layers. This worked well with the primary spacing parameter being pitch: the brown enemies, being large and tough, represented the *low*; the yellow enemies, being small and weak, represented the *high*; and the red enemies, being somewhere between the other two in terms of size and strength, represented the *medium*.

As was seen in Level 1 (Chapter 5), melodic lines are potentially difficult to use when an overall goal of the music is to represent intensity or danger. Therefore, in order to convey motivic information, the layers representing the three enemy types were composed to feel somewhere between a melody and a riff. In this way, it was hoped that these lines would be thematic enough to create a memorable connection to the enemy types, while being rhythmically strong enough to add to the overall perception of danger in the music (see Figures 7.6–7.8 and DVD items 16–18). All three layers followed the same basic harmonic progression so that they would be able to play in any combination and not cause any unintentional dissonance.

Figure 7.6 Extract from the yellow enemy layer in Level 3.

Figure 7.7 Extract from the red enemy layer in Level 3.

Figure 7.8 Extract from the brown enemy layer in Level 3.

The three *game state layers* made up the remainder of the music for Level 3 and changed as the enemies attacked. These layers mainly comprised harmonic and rhythmic elements and could arguably be thought of as including information about the current wave state, but this was information that was also presented visually through the user interface and was not being tested in this level.

The first of the game state layers was the base layer which formed the foundation of the music. This layer contained a simple drum pattern, bass and synthesiser instrument playing arpeggio lines (see Figure 7.9 and DVD item 19), which played the entire time underneath the rest of the music. The remaining two layers, known as the *danger layer* and the *calm layer,* represented the two distinct gameplay states when enemy waves were or were not attacking the player's base. These layers saw a departure from the way in which the vertical layering had worked in the tests up to this point. Whereas, in Levels 1 and 2, layers were always added on top of previous layers and not subtracted unless all higher layers had been removed, these two layers were turned on and off independently of any other layer. This meant that when no enemies were attacking, the calm layer would be added and the danger layer removed, while the opposite was the case while an enemy wave was attacking the player's base.

Figure 7.9 Extract from the base layer in Level 3.

The calm layer contained a cello and a further drum part which added a snare drum and further kick drum to the drum kit already present in the base layer (see Figure 7.10 and DVD item 20). The danger layer added an alternative additional drum kit part and drum and timpani parts (see Figure 7.11 and DVD item 21).

Figure 7.10 Extract from the calm layer in Level 3.

Figure 7.11 Extract from the danger layer in Level 3.

This technique of using two layers, each of which would only ever be played in the absence of the other, could be termed *layer switching* and is necessary to overcome one of the problems with *instrument stacking*. The usefulness of this technique is evident when comparing the additional drum kit parts of the calm and danger layers. The snare and kick drums could not work if the danger layer was simply added on top of the calm layer through instrument stacking. If all three parts were played at once, the snare drum, in particular, would sound odd, playing on beats 2, 3 and 4. Therefore, to create the actual drum pattern of the base layer plus the danger layer, the calm layer must be removed; likewise, to create the actual drum pattern of the base layer plus the calm layer, the danger layer must be removed.

The Music System

As with Level 2, Fmod was again used for the music system in Level 3. The Fmod event itself contained four different parameters to control the musical layers. Three of these parameters were used to control the motivic layers for the yellow, red and brown enemies. In the same way as had been the case in previous levels, these parameters were used to control the volume automation of their designated layer. The final parameter was the *game state* parameter, and was used to control the volume automation on the calm and danger layers. When the parameter was set to its minimum, the calm layer would be at full volume and the danger layer would be inaudible. When the parameter was set to its maximum, the danger layer would be at full volume, and, conversely, the calm layer would be inaudible.

Given that the volume automation for the game state layers was being used simply as a fade in and out, a linear roll-off was used. However, because the volume automation on the motivic layers was being used to convey gameplay information, more attention had to be given to the specifics of the volume automation for these layers. Large enemy groups would be represented by the layers being played at full volume; therefore, the main decision was what volume the layers should be played at to represent a small group of enemies. This decision would undoubtedly be very dependent on the particular composition itself, and could therefore be very different for a different soundtrack. Because of this, a certain amount of experimentation with different combinations of layers playing was used before settling on a suitable value for the volume that represented a small group of enemies. In the end, a value of –8.5 dbs was chosen and was used for all three motivic information layers. This meant that a non-linear volume curve was used for the automation on these three layers, in order to create a smooth transition between volume states, as can be seen in Figure 7.12.

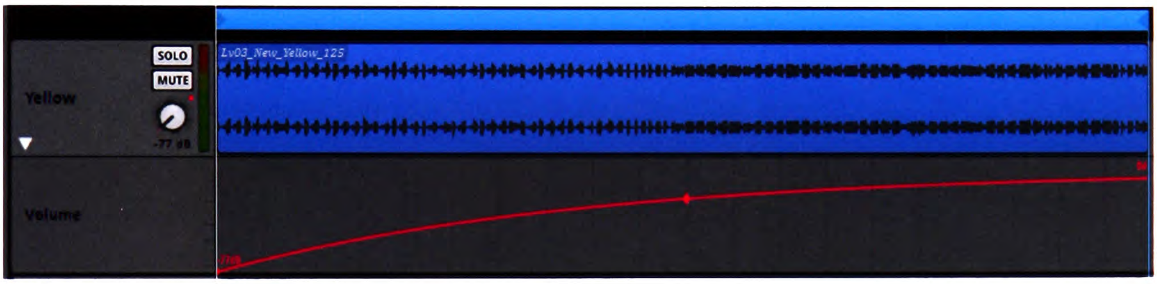


Figure 7.12 Volume automation for Level 3 motivic layers.

Potential Findings

It was believed that players would be able to tell the difference between the three motivic information layers that represented the three different types of enemy, and it was thought that the use of the layer-spacing techniques discussed in Chapter 6 would contribute to making this the case. It was believed that the yellow enemy layer would be the easiest for the players to hear as it contained the highest pitched instrument of the three motivic layers. This would then decrease through red, with brown being the hardest to hear as it contained the lowest pitched instrument. To what degree this issue might be apparent and affect players may depend greatly on what kind of speakers or headphones the players listen to the game with, a possible drawback that was discussed in Chapter 4.

It was believed that the most difficult aspect of the test would be the task of predicting the size of the different enemy groups. As this was achieved by listening and deciding whether the motivic information layers were being played at a loud or quiet volume, this aspect of the test relied heavily on comparisons within the soundtrack and between different waves. Because of this, predicting wave sizes would probably be particularly hard to get right during the early waves of the level. Finally, it was expected that the same pattern of players doing similarly well in the micro- and macro-tasks as they did in their predictions would be present again in Level 3, but it would be useful to have this reinforced in this final test level.

For a video compilation of extracts recorded from Level 3, see DVD item 22.

To play Level 3, see DVD Level 3: test executable folder.

RESULTS

A total of 12 players took part in this final test. This number is slightly fewer than in previous tests as it was difficult to retain players for a third time. Table 7.1 displays the basic results for Level 3. The data is presented in a slightly different way from previous tests. As was mentioned in the Design section above, the first three digits of the results represent the actual enemy composition that was present in the wave, while the second three digits are what the player predicted. Of these three-digit numbers, the first digit represents the yellow enemies, the second represents the red enemies and the third represents the brown enemies. A "0" means no enemies (i.e., an absent enemy group), a "1" means a small group and a "2" means a large group of enemies.

Similarly, this could be thought of as "0" meaning no motivic layer was played, "1" meaning the motivic layer was played at low volume, and "2" meaning the motivic layer was played at high volume – or, in the case of the predictions, this was the player's interpretation.

Table 7.1 Each player's results for Level 3

	Wave 1	Wave 2	Wave 3	Wave 4	Wave 5	Wave 6
Player B	011 010	120 001	111 002	201 212	121 121	101 101
Player C	022 012	111 211	101 101	021 011	102 101	210 110
Player D	112 200	102 001	201 020	022 012	220 000	210 110
Player E	111 000	101 111	201 210	112 102	220 221	110 110
Player F	120 120	220 110	112 102	202 102	210 210	111 112
Player G	220 211	111 120	120 120	201 211	112 121	102 120
Player I	202 010	120 010	112 200	121 211	101 120	011 020
Player L	022 010	111 002	110 120	120 102	211 201	220 211
Player M	210 210	211 211	012 022	110 120	011 012	121 112
Player Q	101 020	022 002	021 020	121 200	120 002	111 000
Player R	022 110	102 022	012 011	112 112	201 222	111 112
Player S	022 001	111 010	211 200	120 021	121 021	012 000

In Table 7.1, the results highlighted in red were auto-accepted by the game. This happened when the player did not press the "submit" button before the prediction time ran out. Some of these auto-accepted results appear to show players simply forgetting to predict altogether: for example, Player E wave 1, where none of the toggle buttons has been selected. Some potentially reveal that a player is running out of time: for example, Player I wave 3, where it is possible that he only had time to make his choice for the yellow enemies. Other cases possibly show players making predictions but forgetting to submit them: for example, Player F wave 6, where he almost predicts completely correctly. Rather than simply removing the most obviously troublesome auto-accepted results (where the player simply had not predicted anything at all, such as Player E wave 1), it was decided that *all* auto-accepted results would be discounted due to the variety and inconsistency of possible reasons for the auto-accept to occur. This decision was also in part made because of the fact that the default setting for the prediction toggles was *off* and, because of this, auto-accepted results have a disproportionately high amount of *absent enemy groups* (see Table 7.2).

Table 7.2 Comparison of auto-accepted results in Level 3

	Absent	Small	Large
Auto-accepted Predictions	38	13	9

Although it was part of the design of the test for the player's predictions to be time-limited, the allotted time was essentially an arbitrary value, chosen in order to keep the test to a reasonable length and to work with the time structure of the waves of attacking enemies to give a sense of urgency akin to the pressure of a real game. The high number of auto-accepted results was likely to have been caused either because the prediction time allowed was too short or possibly because requiring the player to *submit* their predictions was unintuitive and caused many players to forget that this was part of the test. Either way, these are issues caused by decisions made about the design of the test itself and not solely due to the player's actions. The decision to remove auto-accepted results meant that the analysis of this test looked only at predictions that players committed to, not ones that were forced on them by the timer. This should more closely reflect the player's actual interaction with the test level. The main drawback to removing auto-accepted results was that it might remove more *bad* results than *good*. It is possible that players who did not concentrate and pay attention to the timer, or who generally did not have as good an understanding of the test than others, might potentially perform less well in the predictions and generate more auto-accepted results. However, the primary point of the test was not to make a comparison with the music of previous levels, but to continue to explore the concepts of vertical layering and information in music.

Table 7.3 shows the results for the two additional gameplay tasks. The micro-task is displayed in the same way as it was for the results of Level 2 – as an average of the total time both droids were in their respective markers expressed as a percentage of the total length of the test. As with Level 2, there was again quite a large spread of results, which is likely to reflect the varying levels of experience that the players have with RTS games. The macro-task, which was new for Level 3, is displayed as the average time that the player took to spend one set of resources, once the full amount had been gathered. The value here is averaged across all their results to account for the fact that there are different numbers of results per player for the macro-task (as described in the Design section above). As with the micro-task, the macro-task also displays a large spread of results.

Table 7.3 Results for the micro- and macro-tasks in Level 3

	Micro	Macro
Player B	78.2%	9.3s
Player C	86.0%	7.0s
Player D	62.6%	16.2s
Player E	94.9%	2.5s
Player F	92.4%	5.7s
Player G	94.8%	10.5s
Player I	80.9%	16.7s
Player L	57.5%	6.6s
Player M	74.6%	4.9s
Player Q	29.8%	14.0s
Player R	54.4%	38.3s
Player S	52.8%	16.2s

Analysis of Results

For this analysis, the terms correct/incorrect will be used to refer to two different aspects of the players' predictions and it is important not to confuse them. First is the wave prediction overall, which consists of the three enemy types and their group sizes; secondly, each of these enemy types and sizes will be discussed as individual correct or incorrect predictions as parts of the overall wave prediction. Three correct enemy type/size predictions make up a correct overall wave prediction.

Figure 7.13 displays a breakdown of prediction success by three different categories:

1. *Completely correct*: where each of the three parts of the player's prediction for an individual wave matches exactly what was actually present in the attacking wave (both enemy type and size).

2. *Enemy types correct*: where the player correctly identified the presence or not of each enemy type but did not necessarily get the *size* (small or large) correct in the attacking wave. In terms of the music, this meant that the player correctly recognised the presence or not of each of the motivic layers, but did not necessarily get the volume correct.

3. *Two out of three correct*: where two of the three parts of the wave prediction were correct but one could be wrong.

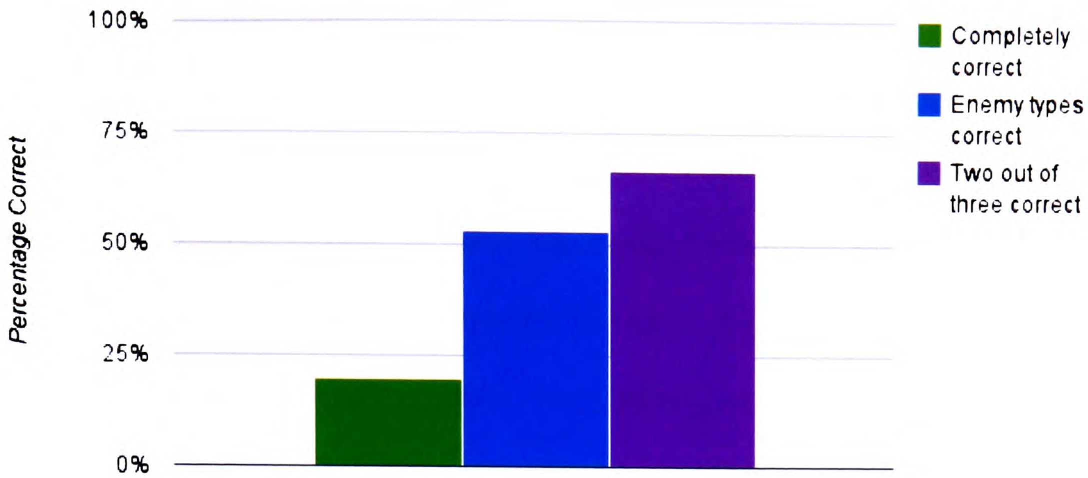


Figure 7.13 Overall prediction success rate by category in Level 3.

Categories 2 and 3 are different ways of looking at the results as *nearly* correct. It should be noted that, while both the "enemy types correct" and "two out of three correct" categories include the "completely correct" responses, neither entirely includes the other. This means that the three categories are *not* simply on a spectrum of *decreasing correctness*, but the latter two categories do contain some overlap. For example, Player C wave 1 would be included in both categories as he correctly predicts both the yellow and brown enemy groups, and the fact that he predicts the large red group as small counts in the criteria for "enemy types correct" and also counts for the one incorrect prediction allowed in "two out of three correct". However, Player M wave 6, for example, only counts for "enemy types correct" as he has got the group sizes wrong for both the red and brown enemies, while Player B wave 1 only counts for "two out of three correct" as his incorrect prediction of absent brown enemies discounts it from having all the enemy types correct. Looking at the results in terms of these categories helps identify which aspects of the predictions players performed best in.

First of all, 19.6% of predictions were completely correct. To put this in perspective, with three possible choices (absent, small, large) for each of the three enemy types (yellow, red, brown) there was a 1 in 27 or 3.7% chance of purely guessing correctly. It would appear, then, that players are not simply relying on guesswork despite the increased complexity of this final test. This was one advantage that this test had over the two previous tests, where in Level 1 each prediction had a 1 in 3 chance of being purely guessed and Level 2 had a 1 in 5 chance.

At 52.9%, over half of the predictions had at least all the enemy types correct. While it could be argued that the number of completely correct predictions is most indicative of how well the players performed in the test, seeing how much higher this category is might indicate that the main difficulty players had in the test was guessing the wave size correctly, rather than recognising the melodies or instrumentation of the motivic layers that represented the individual enemy types.

The final category, "two out of three correct", helps paint an even broader picture of the players' test responses. Without this category, all of the predictions that had only one part of the three incorrect would be ignored, and these are arguably still *mostly* correct and useful to analyse. At 66.7%, this category shows that just over two-thirds of the predictions had at least two of the three parts to the prediction completely correct. The large difference between this percentage and the 19.6% of completely correct predictions could indicate that the difficulty of the test potentially was due to the players having to interpret multiple pieces of information. However, this does not take into account whether any of the individual motivic layers was easier to interpret than others, and this will be examined later in the analysis.

Enemy Group Size Predictions

In terms of the soundtrack, this section will look at the players' ability to tell the difference between volume states of the motivic information layers. Figure 7.14 shows the distribution of predictions for when a small group of enemies was present in the wave. As can be seen in the figure, 59.2% of small groups were correctly identified; 21.1% of the small groups were mistaken for large groups (meaning that the player could hear the layer, but thought it was louder than it actually was); and 19.7% were missed altogether (meaning that the player did not report hearing the layer in that particular wave). Comparing this with Figure 7.15, which shows the same breakdown for large groups of enemies, it can be seen that 56.5% of large groups were correctly identified; 30.5% were mistaken for small groups (this time the player thought the volume of the layer was quieter than it actually was); and 13% of large groups were missed altogether.

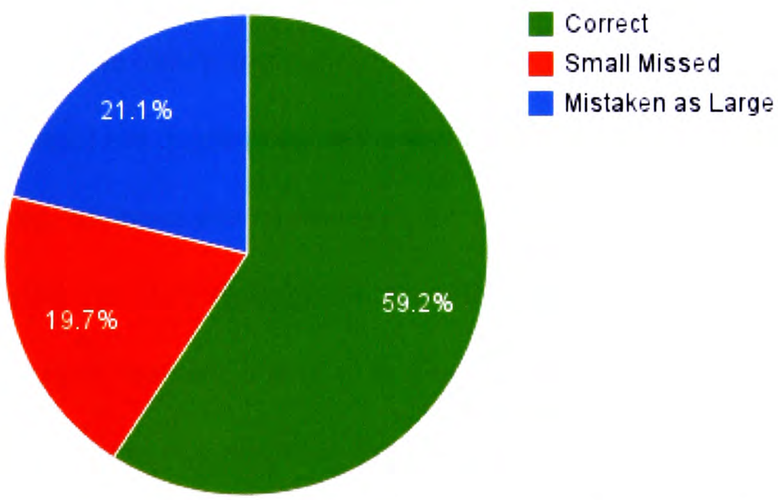


Figure 7.14 Distribution of predictions for small groups of enemies in Level 3.

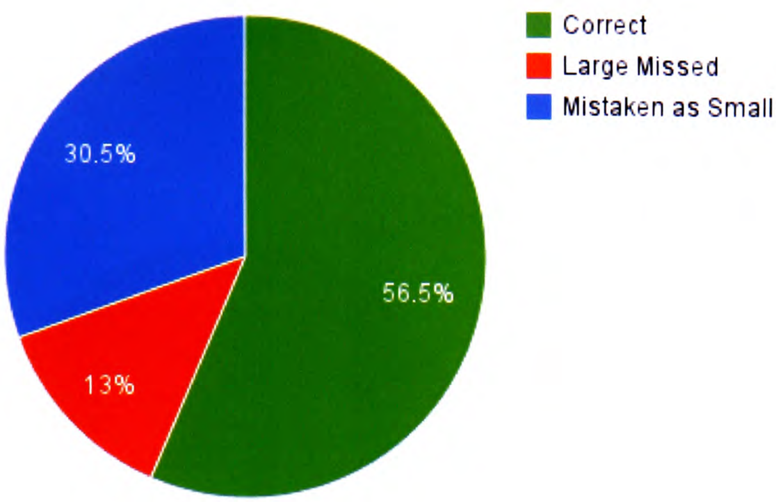


Figure 7.15 Distribution of predictions for large groups of enemies in Level 3.

It would appear from the data that small groups of enemies were slightly easier to identify than large groups, meaning that interpreting the motivic layers as quiet was easier than interpreting them as loud, although this would appear to be counterintuitive. However, the slightly lower correct percentage of large groups could be because of the higher rate of confusion there was with small groups. Overall, these results seem fairly logical: small groups were easier to miss altogether (probably because of the quieter volume that represented them), while large groups were harder to miss but more easily mistaken for small groups. As in earlier tests, where players reported that they thought the soundtracks were not intense enough to represent the numbers of enemies, a similar explanation could account for the results shown here.

Figure 7.16 shows a similar breakdown for the absent enemy groups. These represent when no enemies of a particular colour were spawned in a given wave. These categories are not quite comparable with the small and large groups, as apart from the number of correct predictions, here we see when small and large groups were *imagined* rather than confused for one another, given that no layer was actually played to represent that enemy type. The term "imagined" is used here simply to mean that the player thought they heard a particular layer in the music when it was not actually present.

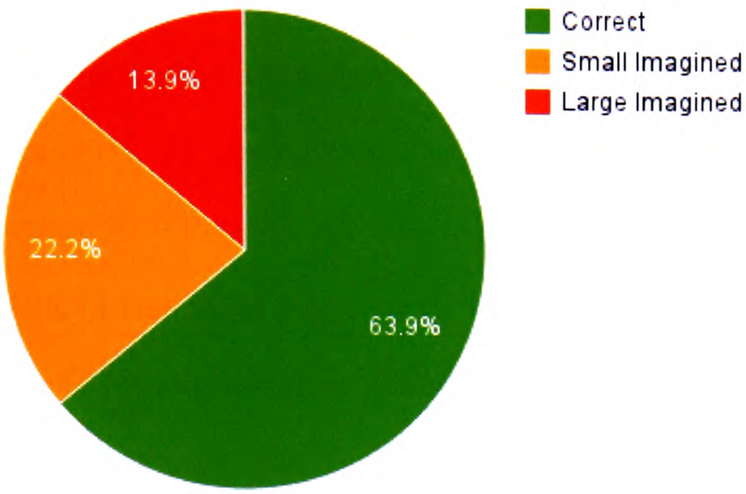


Figure 7.16 Distribution of predictions for absent groups of enemies in Level 3.

In all, 63.9% of absent enemy group predictions were correct. This was unsurprisingly higher than the percentage of small and large groups, given that simply hearing the presence or not of a layer was arguably the easiest aspect of the predictions. Absent enemy groups were imagined as small groups 22.2% of the time, and as large groups 13.9% of the time. Although these numbers are not comparable with the small and large mistaken percentages, they do make sense when compared together. The motivic layers that represented the small enemy groups were played more quietly and therefore it may have been easier for the players to think they heard them in the music, even when they were not present. Given that all of the musical layers followed the same harmonic progression, it is not that far-fetched for the players to think that they were hearing music that was not really there.

It should be mentioned here that there were actually a higher number of small groups spawned during the tests than large or absent enemy groups. There were also fewer absent enemy groups than small or large groups. This was inevitable due to the nature of the test. As noted in the Design section above, there were a number of exceptions made to what wave combinations could spawn, but, simply put, waves had to contain at least two small groups, could only ever include one absent enemy group, and never contain more than two large groups. Because of this, across the 12 test participants, 71 small groups, 46 large groups and 36 absent enemy groups were spawned. It is possible that due to the small number of participants, and therefore results, that a more even spread of spawn sizes would have affected the results, though due to the design of the test it was not feasible to achieve this.

Player Performance over Time

Figure 7.17 displays overall player performance over time. Given the findings of Levels 1 and 2, where it was shown that players improved in their ability to predict enemy waves correctly, the results shown here are somewhat surprising. Both the results for the total number of "completely correct" and "two out of three correct" predictions very slightly decrease with each wave. The "enemy types correct" category follows an opposite trend, on average increasing with each wave.

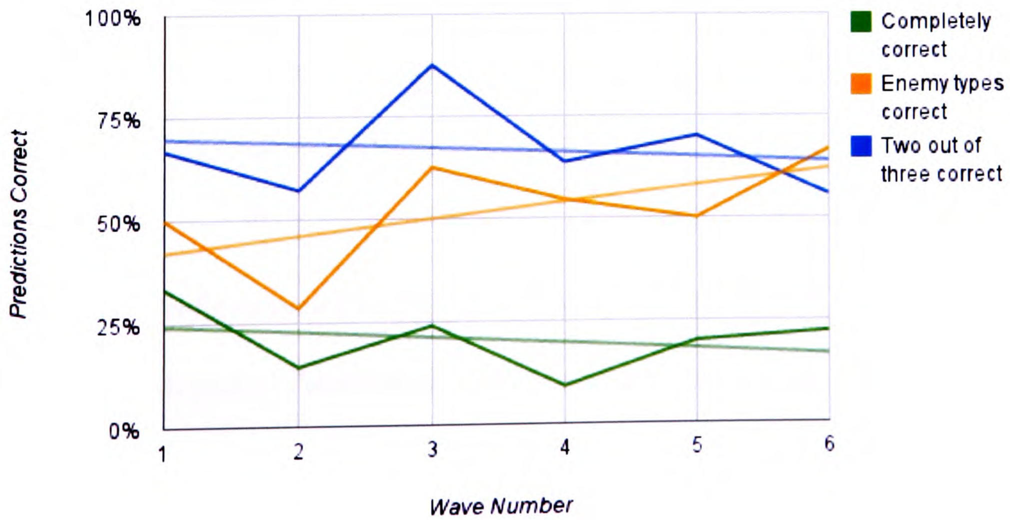


Figure 7.17 Player performance over time in Level 3.

There are a number of possible explanations for this. It is possible that being able to judge the relative volumes of any given layer does not get easier the more the players listen to the soundtrack. In this explanation, the player's ability to create comparisons between waves does not have the same effect for interpreting the relative volumes as it does for learning to recognise the instrumentation and motifs of the layers themselves – as was thought to be the case in Levels 1 and 2. This level includes two dimensions to the information: the players must recognise the motivic information layers and identify whether the volume is low or high.

It is possible that the removal of auto-accepts has contributed to this pattern of performance over time. As many of the auto-accepted results that were removed were from early waves, this has created an uneven number of results per wave. Therefore, there are potentially not enough results to analyse the data on a per-wave basis. Similarly, it is possible that the fact that, unlike in the previous two tests, where there were ten waves each, having only six waves in this level is not enough to gauge player performance over time.

The number of "enemy types correct" per wave does improve over time, however. This might reinforce the notion that players become more familiar with the melodies over time, but do not develop a better way of understanding the volume differences. It is, however, almost impossible to know conclusively whether this is the case, or what the reason for these patterns of results is from the data alone.

Prediction Performance Compared to Gameplay Tasks

Figure 7.18 displays a comparison between the results of the micro- and macro-tasks. Each point on the graph represents an individual player. The point's position on the x axis is determined by the player's performance in the micro-task. This is the average amount of time each of the droids was kept within its marker expressed as a percentage of the total length of the test. The point's position on the y axis is determined by the player's performance in the macro-task.

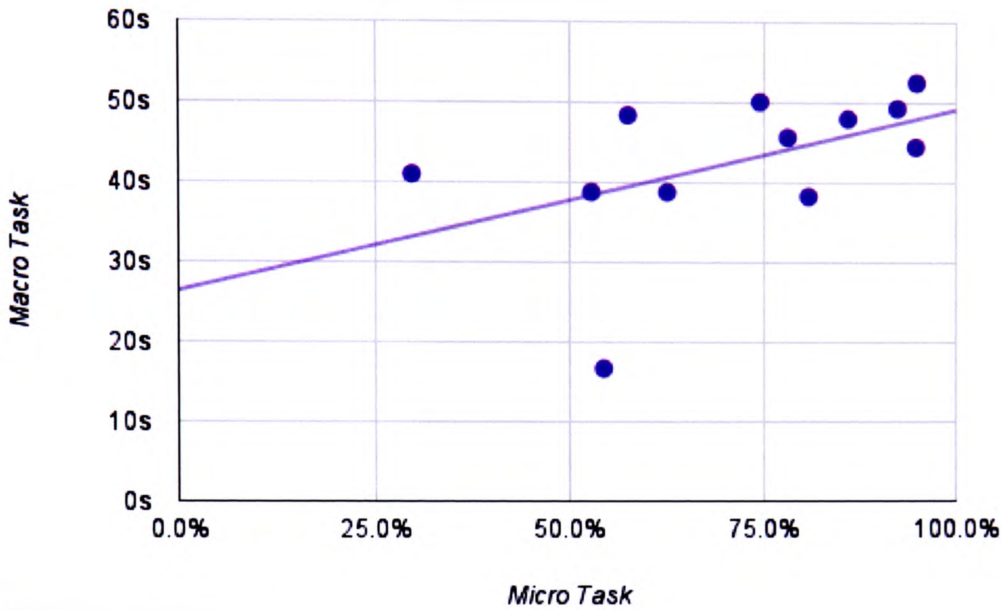


Figure 7.18 Comparison between results of the micro- and macro-tasks in Level 3.

The macro-task value has been converted from the original value shown in Table 7.3 – which was the average time the player took to spend their resources each time the allotted amount had been collected – to the average time the workers were collecting resources per wave. This is essentially looking at the results the opposite way around. This number was calculated by taking the average time for a wave, which was 55 seconds, and subtracting each player's average time taken to spend the resources. This conversion means that the macro results follow a similar format to the micro results in that essentially a high number is better than a low number. This will also help the comparison with the prediction results which will follow. Figure 7.18 simply shows that players who did well in the micro-task, roughly speaking, also did well in the macro-task. While these tasks represent different aspects of RTS gameplay, it appears that players generally display a certain level of gaming proficiency. Therefore, the ability to perform both tasks appears to be related to one another.

Figure 7.19 compares the results of the micro-task with how well the players performed in their predictions of the enemy waves and, by extension, their understanding of the soundtrack. The graph is broken down into each of the three categories used above: "completely correct", "enemy types correct" and "two out of three correct". As seen in Level 2, all three categories suggest a relationship between the player's ability to perform the micro-task and their success at predicting enemy wave types through understanding the soundtrack. Here, the data shows that players who did well in the micro-task also did well in the predictions. Figure 7.20 shows that there is a similar relationship between the macro-task and the wave predictions. This is not surprising given the fact that players who did well in the micro-task also did well in the macro-task. Therefore, there is little reason to expect that the relationship between the macro-task and player predictions would be any different.

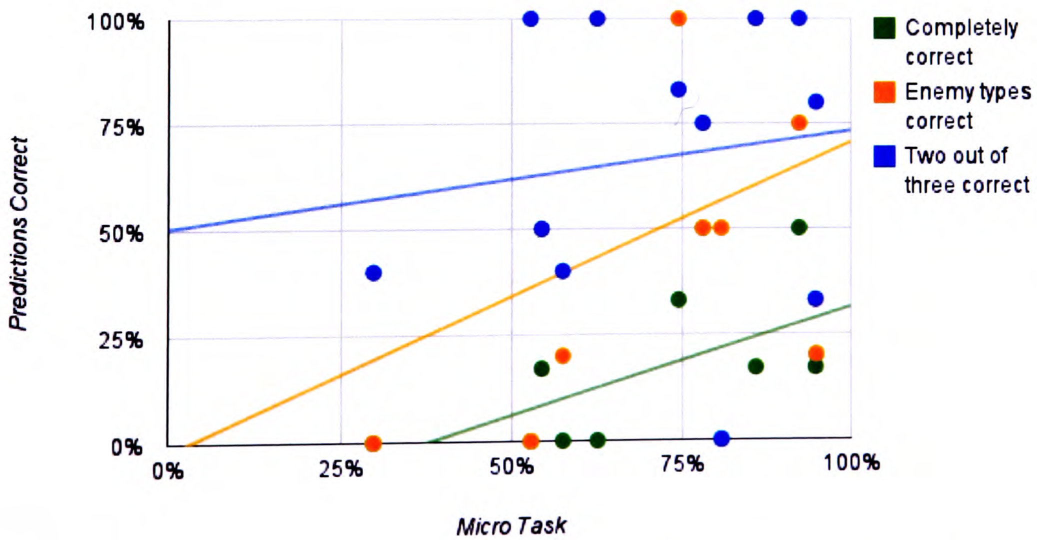


Figure 7.19 Comparison between results of the micro-task and overall prediction success in Level 3.

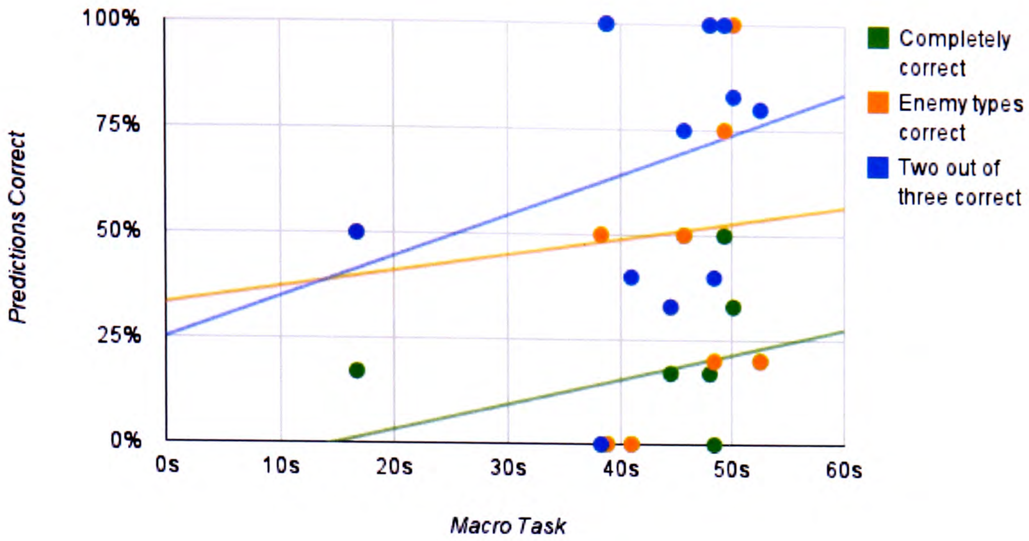


Figure 7.20 Comparison between results of the macro-task and overall prediction success in Level 3.

Success of the Motivic Layers

The motivic layers were the three parts that made up a prediction and not the wave prediction as a whole, as explained above. In terms of the test, this is how well the players predicted the presence and size of specific enemy types. In terms of the music, this is whether the player could distinguish between the three motivic information layers and recognise their relative volumes.

Figure 7.21 displays the percentage of correct predictions for each enemy type and includes both the "completely correct" predictions for each enemy type as well as a category called enemy "identified". This category is similar to the "enemy type correct" category as discussed above, where the presence of the specific enemy type/motivic layer was identified, but the group size/layer volume was not necessarily correct. This category includes all of the "completely correct" predictions. There is no equivalent for the "two out of three correct" category, as this part of the analysis deals with individual enemy types and not the wave as a whole.

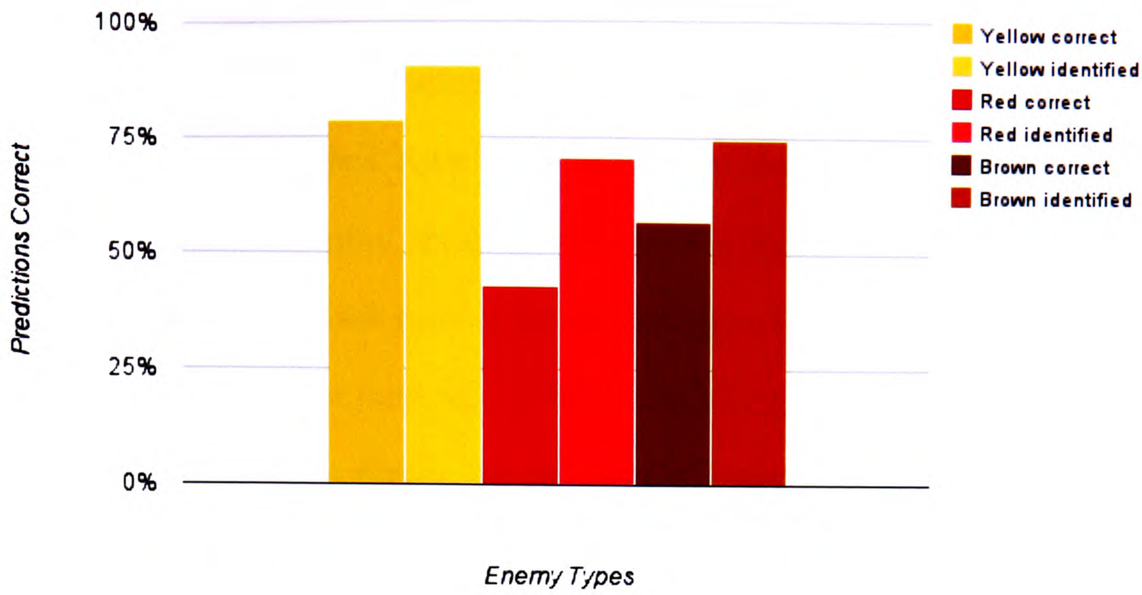


Figure 7.21 Success of individual motivic layers in Level 3.

Yellow enemies had the highest percentage of correct predictions, with 78.4% correctly predicted. In addition, 90.2% of yellow enemies were at least identified as being present, although not necessarily predicted in the correct group size. Brown enemies had the next highest correct response rate, although notably lower than yellow at 56.9% completely correct and 74.5% of brown predictions at least identified as present. Finally, the red enemies had the lowest percentage of correct predictions at 43.1% completely correct and 70.6% at least identified as present. There is no way of knowing for certain from these results alone whether the yellow layer was the most memorable or noticeable and the red layer was the least, but these results suggest that the players were able to recognise the yellow layer most easily and the red layer the least.

The layer that represented the yellow enemies was the highest pitched instrument of the three, so arguably cut through the rest of the music better than the red and brown layers. This could possibly account for why the yellow enemies had the highest correct prediction percentage. There are not many instruments apart from the bass drum that occupy the low frequencies of the music; therefore, the presence of the brown layer might not be affected very much by other layers. It could possibly be the case that certain players' listening set-up made the brown layer harder to hear than others. For example, players' speakers or headphones might not have had a good bass response, making the lower parts of the music less prominent than intended and this may possibly be why the percentage of brown enemies predicted correctly was lower than the yellow enemies. Finally, the mid-range of frequencies, which the red enemy layer occupied, had the most parts fighting for space in the mix. For example, the game state layers could have masked the red layer, causing it to have the lowest percentage of correct predictions. One factor that was not taken into consideration during the compositional process was the fact that the red layer contained more sustained notes and therefore fewer notes overall. This rhythmical simplicity could possibly have made it less prominent compared to the yellow and brown layers. These are, of course, just potential theories to explain the results of how successful each motivic layer was: it is impossible to know for certain based solely on the data.

As noted previously, the two volume states (high and low), which were used to represent small and large enemy groups, were identical for all three motivic layers: 0 dbs for high and –8.5 dbs for low. Even though keeping the values the same for each layer was *fair,* it could be argued that each volume is not perceptually equal. For example, depending on the rest of the soundtrack, –8.5 dbs could potentially sound different for high and low sounds. It is possible, then, that the motivic layers need to have different settings for high and low volume to account for their differing pitches or relative perception in the mix. This will undoubtedly be very dependent on the music overall and therefore it is may be difficult to put forward any general guidelines with regards to this concept.

In order to analyse the success of each motivic layer further, the percentage of correct predictions per enemy type *by group size* has been studied. Figure 7.22 displays these results. Each cluster of three enemy types inevitably shows the same overall pattern as was seen in Figure 7.21. However, what contributed to the aforementioned pattern can now be seen in greater detail. While the specific percentages are higher for the yellow enemies, both yellow and brown have small groups of enemies predicted correctly the least. Absent enemy groups are the highest for yellow, while for brown absent enemy groups and large enemy groups were predicted equally. It is surprising, though, that the opposite is the case for the red enemies, with small enemy groups having the highest correct prediction percentage, and absent enemy groups and large enemy groups both being lower.

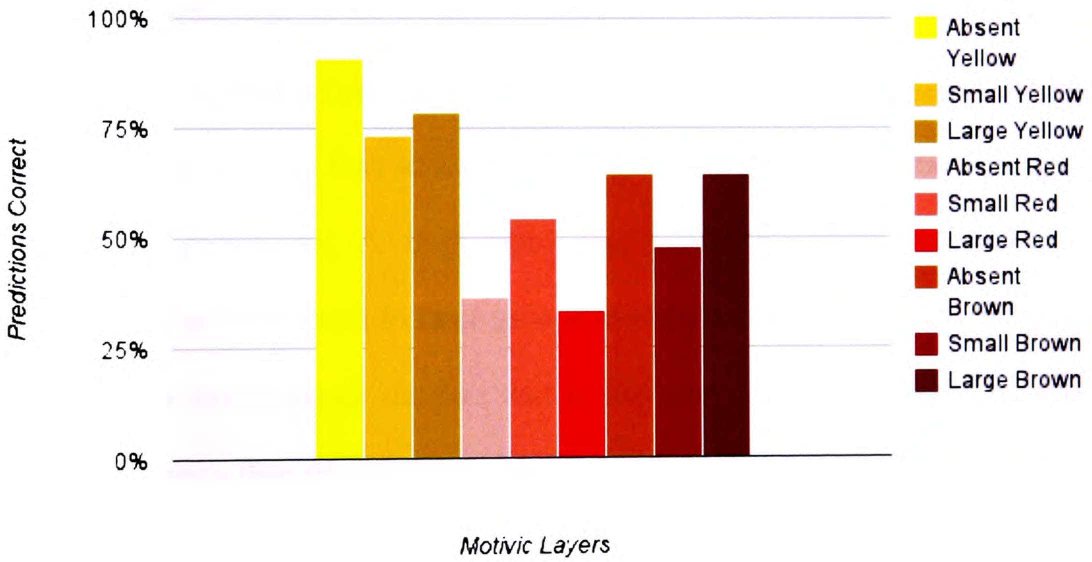


Figure 7.22 Enemy type prediction success by group size in Level 3.

The fact that absent enemy groups had the highest correct prediction percentage for yellow and joint highest for brown was perhaps not surprising, given that predicting absent enemy groups only required players to recognise the presence or absence of the motivic layer. This also matches what was seen in Figures 7.14–7.16, where absent enemy groups saw the highest correct prediction rate when looking at all layers together. In these figures, it can be seen that small and large groups had a similar correct prediction rate: less than 3% difference. Comparing this with the further breakdown by enemy type of Figure 7.22, it appears that the brown layer and, to some degree, the yellow layer contributed towards large enemy groups having a high correct prediction rate (as seen in Figures 7.14–7.16), while the red layer contributed towards small groups having the higher prediction rate.

It is quite likely that the many musical and, more generally, aural differences, such as frequency and note length, between each motivic layer caused the red enemy layer to have the opposite rate of success when it came to group size. Hearing each layer at high or low volume is likely to have a different effect on the perception of the particular layer, due to its musical properties. It is also possible that the player expectations of what a small or large group should sound like, or possibly even how comparatively different they should sound, has played a role in the success rate of the three enemy group sizes. As in previous tests, players may have expected more of a musical change from small to large groups of enemies, regardless of actual gameplay danger. This would match the fact that in this test large groups were mistaken for small more often than the other way round.

Distribution of Predictions for Each Motivic Layer

The final part of this analysis looks at enemy group size predictions for each of the three enemy types individually, including the incorrect predictions. For example, for the yellow enemies: how often were large enemy groups mistaken for small groups or small enemy groups mistaken for absent enemy groups and so on? This will show the players' prediction performance per enemy type or musical layer at its greatest level of detail. This could be thought of as a version of layer confusion – not *between* motivic layers but between the volume and presence of the motivic layers.

This analysis follows on from Figure 7.22 and, because correct predictions per enemy type have already been discussed, this part of the analysis will mostly focus on the *incorrect* predictions. These incorrect predictions have been divided into two types by how *far* from correct they are. For example, for absent enemy groups: "absent" as "absent" is obviously correct, "absent" as "small" is incorrect, but "absent" as "large" is the furthest from being correct. The same logic is used for large enemy groups, where "large" as "large" is correct, "large" as "small" is incorrect and "large" as "absent" is the least correct. Small groups, being the middle group size, have to be treated slightly differently, although the logic is consistent with the absent and large groups. "Small" as "small" is obviously correct, while "small" as "absent" is treated as least correct and "small" as "large" is treated as slightly more correct given that the particular enemy type was at least present in the wave, unlike "small" as "absent". In order to facilitate easy comparison between group sizes, the results in each pie chart in Figures 7.23–7.25 are colour coordinated: green represents correct, blue represents incorrect and purple represents the furthest from correct of the three predictions.

Figures 7.23a–c display the results for absent, small and large *yellow enemy groups*. The figures show that there were no absent yellow enemy groups predicted as large (most incorrect; Figure 7.23a) and that there were no large yellow groups incorrectly predicted as absent (Figure 7.23c). Small yellow enemy groups in comparison, while seeing a lower percentage of size confusion (small as large in this case, as opposed to large as small in the case of large yellow groups), did have a higher percentage of small as absent predictions (Figure 7.23b). From these results, it appears that absent and large yellow groups appear fairly clear to the players, while small yellow groups saw more confusion.

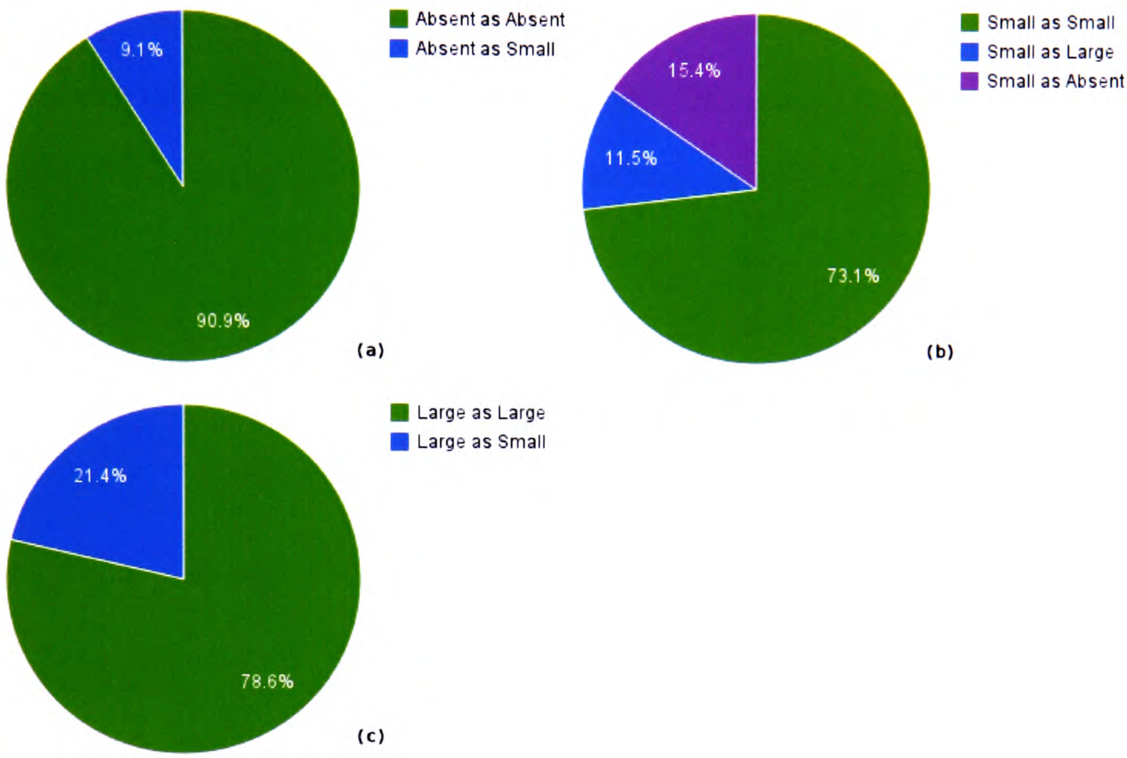


Figure 7.23 a, Predictions for absent yellow enemy groups; b, predictions for small yellow enemy groups; c, predictions for large yellow enemy groups in Level 3.

Figures 7.24a–c show the results for the *red enemy groups*. For absent red enemy groups (Figure 7.24a), the distribution of results appears to be very different compared with absent yellow enemy groups (Figure 7.23a). The percentage distribution is almost equal for each category which might imply that players had difficulty understanding the red motivic layer. Players thought that they were hearing a large red enemy group when none was present almost as often as they got the prediction correct. Predictions for small red groups appear slightly better, although there is still a large percentage of small red groups predicted as large or not heard at all (Figure 7.24b). Finally, for large red enemy groups, there was high confusion with small red enemy groups, given the fact that there is a higher percentage of large as small predictions than large groups predicted correctly (Figure 7.24c). The predictions for large red enemy groups appear notably different from large yellow enemy groups (Figure 7.23c), and the high percentage of large red groups that were predicted as absent is surprising, especially given that no large yellow groups were predicted as absent.

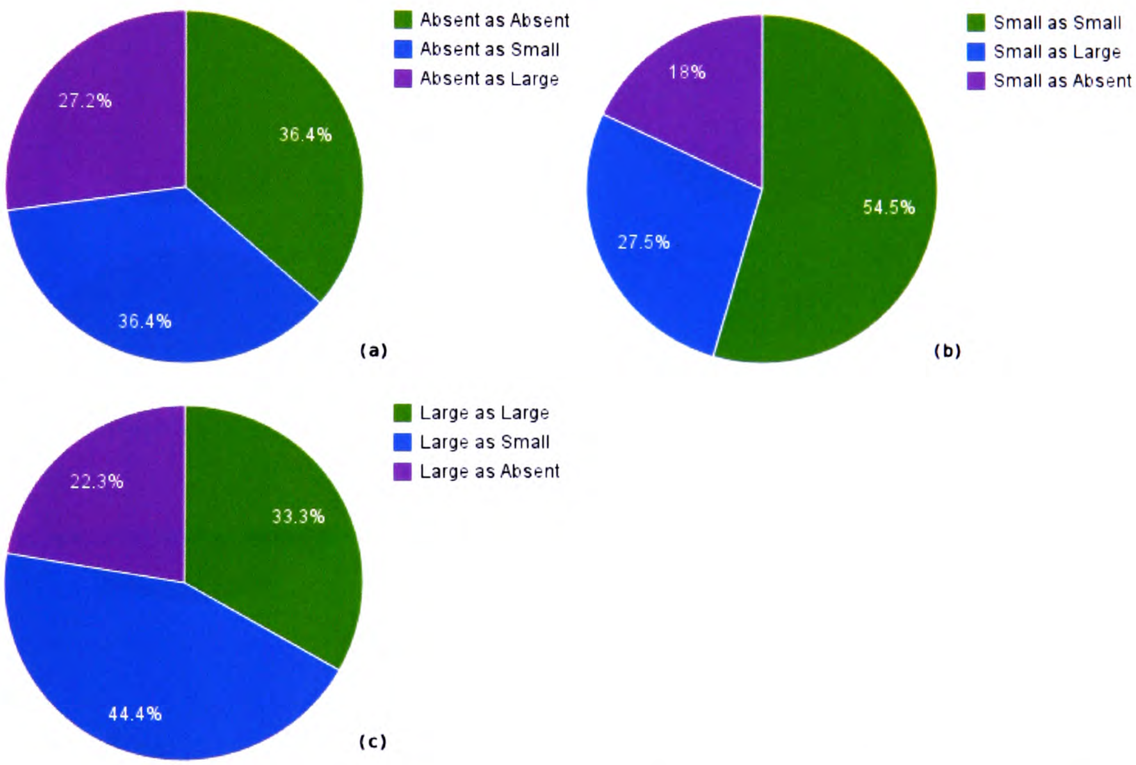


Figure 7.24 a, Predictions for absent red enemy groups; b, predictions for small red enemy groups; c, predictions for large red enemy groups in Level 3.

The final set of pie charts, Figures 7.25a–c, display the results for the *brown enemy groups*. As was seen earlier in Figure 7.22, when looking at prediction success per enemy type, predictions for brown enemies fell somewhere between the success of yellow and red. For absent brown enemy groups (Figure 7.25a), percentages of absent predicted as small and large were much lower compared with absent red enemy groups (Figure 7.24a). Small brown groups, however, saw a much more even and overall higher percentage of the two incorrect categories: small as large and small as absent (Figure 7.25b). Unlike large red groups, large brown groups saw less confusion with their equivalent small groups (Figure 7.25c).

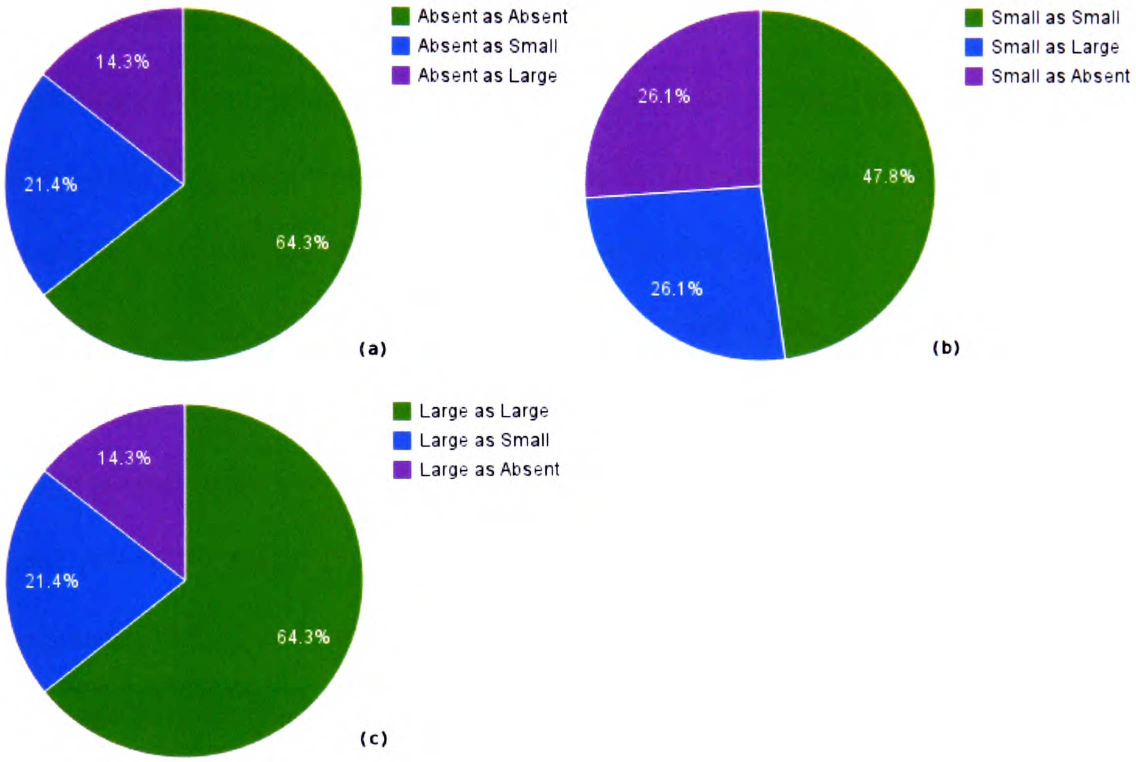


Figure 7.25 a, Predictions for absent brown enemy groups; b, predictions for small brown enemy groups; c, predictions for large brown enemy groups in Level 3.

Overall, this analysis reveals the specifics of the confusion between the particular group sizes for each enemy type. Arguably, it more importantly shows how the areas of confusion are not the same for each enemy type, something that was not visible in earlier parts of the analysis. Much of the enemy group size confusion stemmed from the red enemies – that is, the red motivic layer. Absent red enemy groups, in particular, caused considerable confusion for players. The predictions for this specific wave type were almost equal across the three categories, meaning that player predictions were not much better than pure guesswork when it came to predicting absent red enemy groups. At the same time, these results show just how well players did in understanding the yellow motivic layer. Only 9.1% of absent yellow groups were thought to be small and none was thought to be large (Figure 7.23a). Likewise, while 21.4% of large yellow groups were thought to be small, not a single large yellow group was missed entirely (Figure 7.23c). This would imply that players only had some trouble understanding the volume of the yellow motivic layer but did not have any problems identifying it in the first place.

It is possible that the players may have felt some confusion between the motivic layers in this test. Although it cannot be seen in the results for certain, it could be that players were getting confused between the red and brown motivic layers. This could possibly explain the high percentage of absent as large and large as absent predictions for red and brown enemy groups (Figures 7.24a, c and 7.25a, c). Players may have thought that what they were listening to was one layer, but in reality it was the other.

While the data cannot categorically show confusion between layers, as it could in previous tests, specific examples in the results could potentially show this confusion occurring. For example, in Player G's wave 6 results, absent red and large brown spawn, while he predicts large red and absent brown, as well as predicting small yellow correctly (see Table 7.1). Similarly, in Player L's wave 4 results, large red and absent brown spawn, while he predicts absent red and large brown, as well as predicting small yellow correctly. Finally, the same result occurs for Player Q in wave

5, although he misses the small yellow group. It could well be that the players were listening to one particular layer, but thinking it was representing a different type of enemy, and it would appear that the majority of these motivic layer confusions happened between red and brown. Although it cannot be said for certain, it is possible that this could explain the high percentage of large as absent and absent as large predictions for these two motivic layers.

FEEDBACK

Understanding the Motivic Layers

Players were asked to describe each of the three motivic information layers, using any sort of descriptive terms, whether musical or non-musical, that they thought matched the qualities of the music. It was hoped that this might give some insight into whether or not certain layers were more memorable than others. Players were encouraged to give as many terms as they could, but were not pressured into saying more than they could easily remember.

Player G's description of the yellow layer was "glitchy, chip tunes, plucky, electronic and short, fast notes". His description of the red layer was "distorted guitar and reverb". For his description of the brown layer, the first thing Player G said was "low, deep notes", which triggered him to say that he wanted to add "high" to the yellow descriptions and "middle" to the red. Player G had clearly noticed that the three layers occupied high, medium and low frequencies during the tutorial: "I played the yellow first, then the red, then when I got to the brown I realised the pattern and it was the lowest one." This highlights the importance of how music is presented to the player for the first time. If the layers had not been in this order, then the relevance of their pitch might not have been noticed. In the case of these test levels, the layers are presented in the form of tutorial examples, but in a real game their presentation for the first time could be much more subtle, but still achieve the same effect.

Player E described the layers in the opposite order to how they were initially presented. He described the brown layer as simply "bass"', the red layer as "a kind of middle guitar" and the yellow layer as "a top melody". It is very clear that, like Player G, Player E noticed that the layers were designed to fit into high, middle and low frequencies. When asked to describe the motivic layers, Player E very interestingly noted how he found hearing the layers in the tutorial very different from how they sounded in the actual context of the level: "I could tell the three parts in the preview, but it's different to listening in game … [In the tutorial] you think, 'okay, that's very obviously yellow, that's very obviously red' ... Then when it gets to [the game] it's like 'Is that red? I can't hear red!'" This reinforces the notion (as discussed in the Results section above) that it was potentially the game state layers of the music that made the red layer the hardest for players to hear. It is likely that the rest of the mid-range frequency instruments in the soundtrack masked or blended with the red layer, making it harder to hear.

Difficulty of Hearing the Motivic Layers

Players were asked further about their experience in distinguishing between the motivic information layers and whether any stood out as easier or harder to hear in the music. Player G stated that he found the yellow layer to be the most recognisable as it was the highest pitched layer. Player E thought that the red layer was the hardest to hear and agreed with Player G that the yellow was the easiest: "The yellow is very distinctive, so I didn't have a problem with that. The brown was fairly distinctive whereas the red one blended in a bit more, I think."

It is possible, then, that the best place for motivic information in a vertical layering soundtrack is in the high frequencies of the music. It seems as though the mid-frequencies of the music were over-saturated, potentially causing the red layer to be less clear to the players. Although this seems to be the case for this particular level and soundtrack, it is probably not a hard and fast rule, but very dependent on all other aspects of the music. Had there been, for example, more low drums and percussion or another bass part, the brown layer might have been equally problematic. It is simply the case that the majority of the game state layers occupy the mid-range frequencies.

Multi-tasking

Players were asked how they found performing the three tasks (predictions, micro- and macro-tasks) at the same time. Player G stated that he found the macro-task quite hard as he was mostly busy concentrating on the other aspects of the test. He thought that it would have been easier if there had been some sort of sound to alert you to when you needed to spend the resources. This is similar to something that Player N brought up in his feedback for Level 2, with regards to the micro-droids making a sound when they left their markers. Given that the players are participating in game audio research, it is not surprising that they are thinking in terms of functional uses for audio. Although sonifying the gameplay tasks was not the research aim of these levels, as the focus was on conveying a specific aspect of information solely via the musical soundtrack, it is very interesting to see players thinking in these terms.

Player M noted how his understanding of the soundtracks was hampered by having to perform the two gameplay tasks. While he stated that he thought the layers individually were "very distinct", he thought that, when it came to performing the gameplay tasks, he found listening to and deconstructing the tracks "more challenging". Player E is an RTS player and therefore did not find performing the three aspects of the test too demanding, but did note how the multi-tasking added to the difficulty of the level: "It was more like playing a traditional RTS really ... It makes it harder having to do everything at once ... It makes it more challenging." These comments justify the inclusion of the multiple tasks for the player to perform, not only to make the level more game-like, but also to add to the difficulty and create a more complex experience for the players.

Other Comments and Feedback

Players were given an opportunity to comment on anything else about their experience of playing test Level 3. Player E commented on how the motivic layers fade in: "there was one [wave] where I felt like it was a small yellow and it turned into a large yellow ... I had it ticked but then it got louder and I was like 'ah, now that's a large'". This is an inevitable issue when trying to add layers smoothly to the rest of the music with a fade-in but also using volume to convey information. A possible solution could be not to fade in the motivic layer, but instead use some form of beat synchronisation to add the layer at its target volume immediately but in a musically and rhythmically consistent way. This does have the potential to be jarring for the listener if the music changes too abruptly and is arguably a much less elegant solution than fading the layers in. As well as this, transitions have to happen on beats or bars rather than right away. This is one of the key advantages of using volume control in vertical layering: the control of the music can be independent of musical time and rhythm constraints. There are, of course, advantages and disadvantages to both methods of introducing layers, but it is most likely that the time limitations of the test might have exacerbated the issue. Players knew that they needed to make predictions as quickly as possible, thus forcing them to start interpreting the layer while it was still transitioning. In another context – one that did not include the pressure of limited time – the problem might not have been noticed at all.

Player G brought up an interesting point about the difficulty of remembering the motivic layers for the course of the test. He found that the calm layers that played between enemy waves caused him somewhat to forget what the actual motivic information layers sounded like. This was not something that was anticipated during the design of the level, but could in fact account for the decreasing rate of prediction success that was noted in the Results section above. This phenomenon was not something that had been present in earlier levels given that, apart from the base layer in each test, no game state (non-information) layers were used. The idea and effects of disrupting a player or causing them to forget the music they are trying to remember could itself be an interesting area of research.

CONCLUSION

The goal of test Level 3 was to explore the second method of controlling a vertical layering soundtrack that was introduced in Chapter 5: using a small number of individual layers but using the volume of these layers to convey an additional dimension of information. These layers contained motivic information inspired by the use of musical motifs to represent types of enemy in *Left 4 Dead 2*. Therefore, the motivic layers of Level 3 each represented a different type of enemy, and the size of the enemy group was represented by the layer's volume. The characteristics of the three enemy types were used to inspire the composition of their respective motivic layers.

The way in which enemies spawned in Level 3 was changed to allow for this additional complexity. The level featured three different types of enemy – yellow, red and brown – and these could attack the player in many different combinations, which created a lot of variation and meant that it was very unlikely that players could purely guess all of their wave predictions correctly.

As well as the motivic information layers, the soundtrack of Level 3 also featured game state layers that reflected when the enemies were and were not attacking the player. Although these layers could be said to contain information, this was not being tested in this level. These three layers saw a departure from how layers had been used non-subtractively up until this point. In this level, while the base layer persisted throughout, the calm and danger layers were added and removed independently of any other layer, reflecting the waves of enemies that attacked the player's base. Finally, an additional gameplay element, the macro-task, was added to Level 3 to mimic the macro-management that is present in RTS games. This created a more game-like experience, added further distraction and provided another metric with which to compare players' musical comprehension.

With a total of 19.6% of enemy waves predicted completely correctly, and a 3.7% chance of being able purely to guess, it would appear that the players of Level 3 were making informed predictions of the composition of enemy waves based on their understanding of the soundtrack. A total of 52.9% of wave predictions correctly identified all enemy types, although not necessarily the group size. The difference between this figure and the percentage of completely correct wave predictions might suggest that players had greater difficulty identifying the volume of motivic layers than recognising their other musical attributes as was originally predicted. As each prediction contained three parts (the three enemy types), the results also show that

66.7% of wave predictions were two-thirds correct (two out of three parts correct) which is arguably still mostly correct.

By analysing the results in detail, it can be seen that small groups were the most easy to miss, potentially because of their representation by a low volume, while large groups were harder to miss but more easily confused with small groups. Absent enemy groups were the easiest to identify as they only required the player to realise the motivic layer was or was not present and not identify the volume. Not surprisingly, absent enemy groups were thought to be small more often than large.

Player performance over time differed from previous tests. While recognition of motivic layers appeared to get slightly better over the course of the test, overall predictions saw the opposite pattern. It is possible that interpreting relative volumes does not get easier for players the more layers they hear, unlike learning and recognising the layers themselves. As was raised during player feedback, it is also possible that hearing game state layers caused players to forget the information layers over the course of the test, although player performance over time was potentially inconclusive due to the lower number of waves and therefore results for this test.

With regards to the success of predicting the individual enemy types and therefore the understanding of the motivic layers, the yellow enemies had the highest percentage of correct prediction, followed by brown enemies (although quite a bit lower), with red enemies being the least often correctly predicted. This was possibly because of the amount of parts fighting for space within the mid-range frequencies, but also could have been because of the musical characteristics of the red layer, specifically how it contained more sustained notes than the other two motivic layers. With the yellow layer containing the highest pitched part, it did not have to compete with the other game state layers that contained predominantly mid- and low-pitched instruments. It could also be possible that players' individual listening situations and hardware affected their perception of the three motivic information layers. It should be noted that the results of the test cannot reveal for certain whether players found it harder to hear the motivic layers amongst the rest of the music or whether anything about the layers' musical characteristics made them harder to memorise or whether both factors played a part and to what extent.

Based on the analysis of enemy type by group size, it appears that small groups of enemies, meaning low volume motivic layers, were predicted least accurately for the yellow and brown enemies. The opposite, however, was true for the red enemies, with small groups predicted the most accurately. It is possible that low and high volume levels may well have a different effect on the perception of musical layers depending on their frequency content.

In terms of the analysis of enemy group size confusion, and therefore volume misinterpretation, it has been shown that much of this confusion stemmed from the red layer. The interpretation of absent red enemy groups caused particular confusion where predictions for absent as absent, small and large were almost equal, meaning that players' predictions were not much better than guesswork. However, this part of the analysis also showed how well players interpreted the yellow motivic layer: there was very little confusion between absent, small and large enemy group sizes.

Much of the player feedback reinforced the findings of the data gathered from Level 3. Players reported finding the yellow motivic layer the easiest to identify and red the hardest. The frequency content of the game state layers may have caused the red motivic layer to be less prominent in the mix. Some players clearly noticed the high, middle and low structure of the motivic layers and this may have made the soundtrack more intelligible to them.

8

Conclusion

The research presented here has used the genre of real-time strategy games as a context in which to explore the use of a vertical layering soundtrack as a tool to convey gameplay information to a player. A series of three, purpose-built, RTS-style game levels were developed in order to explore and test this function of game music on a small group of video game players.

The methodology used in this research has not attempted to draw definitive conclusions about the usefulness of vertical layering in conveying information to a player, but this initial exploration has aimed to draw attention to various aspects and design considerations that relate to this use of music in games and potentially to open up avenues for further research. Given the small sample size of players, it should be made clear that any conclusions drawn from this research are put forward tentatively. While the players who took part in this research may or may not be representative of the wider gaming population, the information gathered from this study may still be useful in raising topics for further discussion.

FINDINGS

From the results and analyses of the three test levels, a number of observations have been made and, as a way of concluding this research, are put forward here as potential considerations for designing vertical layering music that is to be used as a source of gameplay information.

As was shown throughout all three test levels, players appeared to be able to learn individual layers of a soundtrack, recognise them during gameplay and respond accordingly to the information they conveyed. Players exhibited the ability to multi-task during the tests: they were able to listen to and interpret the soundtrack while they performed the additional gameplay elements that took the form of the micro- and macro-tasks. Although it is unclear whether performing the gameplay tasks had a negative impact on the players' prediction performance, players were able to concentrate on both playing the game and listening to the music and there appears to be a link between how well a player did in the predictions and the gameplay tasks. On average, players did roughly as well in the one task as they did in the other.

The way in which a designer presents a vertical layering soundtrack to the player for the first time may be important. The order in which the player hears the layers is likely to have an effect on their overall perception of the music. As was seen in the test levels, as well as in player feedback, it seemed as though the order in which players heard each layer during the test slightly affected their overall performance.

Players appeared to be able to improve their understanding of the soundtrack throughout the course of a play session, potentially by continuous reassessment of their knowledge of the musical layers. However, as was seen in test Level 3, there may be certain aspects of vertical layering, such as volume manipulation, that may not become easier to interpret the more the music is listened to.

When designing a vertical layering soundtrack that aims to convey information, it is important to bear in mind that players will always be influenced by their prior experiences of music. Therefore, a vertical layering soundtrack should aim to convey its information by leveraging the right balance between the expected player *preconceptions* and the designer's *intentions*. It is unreasonable to ask players to put aside their expectations and only think in terms of the *learned* meaning of the music.

Understanding player expectations of what various music-to-information mappings should sound like is therefore a crucial part of designing information into a soundtrack. Because of this, testing that these expectations match the actual music needs to be a part of the game development process if music is to play a greater part in a game's information system. Players might forgive slightly mismatched music if it has no other effect on the game, but if it is important to the gameplay, music-to-information mapping needs to be as accurate as possible.

It is therefore important that the structure of the layers of a vertical layering soundtrack (both their composition and control) must match the structure of the information. This research has suggested that vertical layering soundtracks are particularly suited to conveying information that takes the form of continuous messages and, as such, each degree on the possible scale of information must be representable in the music.

Two methods of controlling layers have been identified within this research: *on/off states*, where the music is composed in enough individual layers to convey all degrees of information and where these layers are simply either on or off in the mix; and *volume control*, where the music is composed of fewer, potentially more complex musical layers, but which have their volume manipulated to affect their perception within the overall soundtrack. These two methods of controlling musical layers illustrate different ways of representing points on a scale of information. On/off states, though, are limited to representing only as many specific points on an information scale as there are individual layers, while volume can represent almost limitless degrees of information, although these would be almost certainly far less clear to the listener. Which method is most suitable may be highly dependent on the type of information being conveyed.

The terms *motivic information* and *motivic information layers* have been used as a way of describing particular layers that represent distinct pieces of gameplay information. In Level 3, these were used to varying degrees of success to represent the three different types of enemy that attacked the player's base. When combined with the concept of volume control, this type of musical layer has also been used to convey *two* dimensions of information. In the example of Level 3, motivic information layers not only represented the presence of a specific enemy type, but also indicated to the player, by their volume, in what size of group those enemies were attacking.

It is potentially important to consider the frequency content of layers within a vertical layering soundtrack. It is possible that certain layers may be harder to hear within the mix if the rest of the music, or the game's audio in general, is over-saturated with any particular frequencies. Likewise, more specifically, when using layer volume to convey information, it is possible that the relative frequency content of a particular layer will affect its perceived volume in comparison to the rest of the music.

The most suitable place for motivic information may be in the high frequencies. Although it will be highly dependent on the specifics of each individual composition, important layers that need to be perceived by the player should be prominent within the mix and not be fighting other parts of the music or audio for space. This research has not studied this aspect of vertical layering in any detail, and so does not make any firm claims in this respect, but it may be an interesting area for further research.

The term *musical representation of information* has been put forward and used as a way of referring to the particular musical qualities that are featured in the layers and that therefore are added or subtracted with each layer in order to convey the specific information of that particular piece of music. The compositional process of a vertical layering soundtrack should endeavour to include the correct musical representation of information in each layer so that the combined music of each state of the vertical layering soundtrack matches the gameplay information it is attempting to represent.

This research has focused almost exclusively on conveying information to the player about numbers and types of enemies, and has therefore used a musical representation of information that could be thought of as musical intensity and danger. It is believed that vertical layering is well suited to this type of representation, but there are likely to be many other qualities of music that can be harnessed to convey other types of information. As an example in real-time strategy games, the player's economy (the collection and spending of various resources) may be conveyed through a musical representation of information using musical qualities that might create a sense of positivity or negativity, but could also be represented by qualities of movement or pace.

As has been seen in the three test levels of this research, adding more layers to a soundtrack is a simple but effective way of increasing its perceived intensity, but adding these layers inevitably creates more complexity which, in turn, makes hearing and understanding individual layers more difficult. However, designers should endeavour not to make their vertical layering soundtracks too simple or their information mappings too literal. As was seen in Level 2, it is possible for players to deconstruct an overly simple soundtrack and, while gaining the necessary information, they could potentially spoil the music's other, non-informational, aesthetic qualities.

The concept of *instrument stacking* has been put forward as a way of changing the perception of one layer by adding another layer that features the same instrument as the first layer and therefore adds further notes. Likewise, the concept of *layer switching* has also been suggested as a way of circumventing some of the negatives of instrument stacking. Two further layers featuring the same instrument as a first layer can be used to create two different versions of a particular musical line by switching between the second two layers.

Having individual layers of a soundtrack clearly identifiable is in many cases an important aspect of conveying specific information through vertical layering. Therefore, a potential method of musical layer spacing has been suggested using the principle of high, medium and low categories. By choosing a musical parameter such as pitch, volume or note frequency, composers can differentiate each musical layer by making the musical parameter in each of these layers high, medium and low.

Musical training of any form will be likely to have an impact on a player's ability to interpret information from a vertical layering soundtrack. The fact that some players may have more difficulty than others in obtaining information from a soundtrack should not be viewed as a limiting factor. Difficulty is a defining aspect of games, and interpreting the soundtrack could be considered part of the game's challenge. This should be taken into account when designing for vertical layering, but should not deter game designers from using music as information in their games. Understanding the music could be thought of as another aspect of mastering a game and therefore a deliberate part of the game's design. Arguably, this is where the use of sound to convey information in games has an advantage over other applications, such as those studied in the field of auditory display, where the purpose of the information conveyed by the audio is to be as clear as possible.

While this research has been contextualised within the genre of real-time strategy games, many of the concepts discussed are potentially applicable to almost any type of video game. It may well be the case that certain aspects of a game's design lend themselves towards this informational use of music, and it is likely that games of an analytical nature rather than reflex-focused gameplay would be able to make best use of conveying information through music. It may even be the case that players of these types of games would be more accepting of this concept.

REFLECTIONS AND FUTURE DIRECTIONS

This research has focused on a functional use of music which aims to convey gameplay information to a player, and it is possible, particularly within the framework of vertical layering, that the functional nature of this use will have an impact on the composer's artistic freedom. However, any form of creativity is subject to limitations, and composers of video game music have always worked within technological and artistic restrictions. While this issue has not been the focus of the research presented here, it is certainly an important consideration and could be the focus of further research.

A number of improvements to the methodology used here could potentially be considered for any future research in a similar vein. The complexity of the individual test levels meant that some players may have found it difficult to adjust to the testing situation quickly enough for their results to reflect their actual understanding of the musical soundtracks and gameplay information without being affected by the pressure of having to interact with an unknown system. While it was believed that keeping the test length short (and therefore the demand on the participants at a reasonable level) would create a positive attitude towards the research during the process, as well as increasing the likelihood that players would participate in further tests, it is possible that additional training of some kind before the test began would have been beneficial to creating a more normalised set of results. This would have meant that fewer results would have been affected by the testing environment. This training could have taken the form of dummy exercises in which players interacted with the test, and could even have used placeholder music so that this was not to allow the players more time to become familiar with the music, but just to acclimatise to the testing situation. It is possible that more focus on finding players willing to participate in longer, more in-depth tests would have been necessary, but this in turn might result in a sample less representative of the wider gaming population.

While the conclusions of this research have been presented as considerations for the use of vertical layering music to convey information based on the findings within the three test levels, future research could build on this initial investigation in order to develop more concrete design principles for this use of music in video games. Although this research was contextualised within a specific genre of video games, it is possible that an even more definitive context could prove fruitful, such as a specific scenario within a specific game. This research has arguably been affected by the fact that it was investigating two variables: the *intent* (using music as information) and the *means* (the actual music used). If one or other aspect were more definitively set – for example, the concept was to solve a specific design problem within a specific game – while this may provide outcomes less transferable to other contexts, it would be likely to generate more detailed knowledge within its limited scope. However, the way in which this research has been structured and the methodology used were necessary due to its role as a preliminary investigation into an area of game audio that has not as yet received much academic attention. Further research within a more specific context to solve specific problems through the use of music may now be feasible following on from this research.

Further research in this area could potentially benefit from drawing on other related fields, which this research has only tangentially touched upon. Semiotics, and specifically musical meaning that takes a more player-centric focus, could be beneficial as it was shown in this research that designers cannot simply rely on players to put aside their preconceptions of music and focus solely on a learned meaning. Likewise, further research into music production, and specifically mixing techniques, would almost certainly be beneficial in adding further insight into some of the layer spacing techniques that were put forward in Chapter 6. Effects processing of individual musical layers to convey information is a potential area for further exploration. As was discussed, vertical parameters of music are time-independent and, as such, timbre is a suitable candidate for the conveyance of gameplay information.

This research has aimed to highlight key aspects of the use of vertical layering music to convey information to a player, and has potentially opened up avenues for further research in this area. Drawing inspiration from the work discussed in Chapter 2 on auditory display studies, one such area of further study could relate to the concept of the *exploratory* function of certain auditory displays. If players were given the ability to manipulate the playback of a game's vertical layering soundtrack in some way – for example, allowing them momentarily to reduce the volume of non-information layers to hear the actual information layers more clearly (in a sense like a heightened version of the *cocktail-party effect*) – could this be a creative way of engaging the player with a game's soundtrack as well as conveying information through it? A feature like this could be completely integrated into the gameplay, being themed to the context of the game itself and perhaps even given a limited activation time and recharge period, ultimately making it an integral part of the gameplay decision-making process for the player. In the same way that a player may have to decide when to use their most powerful spell, they could also have the ability to decide when to manipulate the music to their best advantage.

Appendix 1: Level 1 Tutorial Transcript

Introduction

Greetings, Test Subject, and welcome to Level 1 of the *RTS Soundtrack Experiment* which tests the workings of a multi-layered musical soundtrack.

The level itself is very simple: waves of enemies will spawn and attack your base. The Automatic Flame Turrets placed around the perimeter should take care of most of the enemies, but should they require support, you can use the Hover Tanks provided. Simply *left-click* to select one, and *right-click* to move it or attack an enemy. You can select multiple Tanks by *left-clicking* and dragging a selection box over the desired units.

Please note: the test requires you to take a screen-shot of the results when you have finished, so please make sure the game is running in "windowed" mode before you start. If you are running in full-screen mode, please restart the game now and select "windowed" mode.

Level Tutorial

The goal of this test is to study how well a dynamic soundtrack can aid a player during gameplay. In this level, the musical soundtrack will change depending on how many enemy units are attacking you in the current wave. You will be notified on-screen when the enemies have spawned and a visual countdown timer will start. Before this timer finishes and the enemies arrive, you should listen to the soundtrack and make a prediction of how many enemies are attacking. You do this by pressing one of the three buttons at the bottom of the screen. The predictions can be: 1–3 enemies, 4–6 enemies or 7–9 enemies.

Soundtrack Tutorial

You will now be played an extract of each of the musical layers that make up the soundtrack and represent the different numbers of attacking enemies.

**Example Music*

Start of Test

Once you press the "Continue" button, the level will begin. You will then face ten waves of enemy Tanks. Remember to watch for the notification that tells you when the wave has spawned, and make your prediction of the number of enemies in the wave before the countdown ends and the enemies arrive.

Good luck, and remember that it is not possible to perform badly. It is, after all, the music that is being tested – not you!

Thank You

Thank you for taking part in the *RTS Soundtrack Experiment.*

The data gathered from your participation will be invaluable in helping create a better understanding of the unique relationship between a player and a soundtrack that is present in games.

Once you press the "Fetch Results" button, you will see a print-out of the data from this test. Please take a screen-shot of this and email it to the email address below the results.

Thank you once again. The test is now complete.

Appendix 2: Level 2 Tutorial Transcript

Introduction

Greetings, Test Subject, and welcome to Level 2 of the *RTS Soundtrack Experiment*. This level includes an improved version of the Multi-layered Music System from the first level and features further gameplay mechanics.

The premise of this level is very similar to Level 1, so if you played that level, most of what you are about to partake in will be familiar to you. Don't worry if you didn't participate in the first level, however, as the following instructions will tell you all you need to know.

In this level, waves of enemies will spawn around the edge of the level and attack your base. The Automatic Flame Turrets placed around the perimeter will take care of most of the enemies. If the turrets require backup, you can use the Hover Tanks provided. These are controlled using your mouse: *left-click* to select one, and *right-click* to move it or attack an enemy. You can also create a selection box by *left-clicking* and dragging the mouse; this can be used to select multiple units.

Please note: the test requires you take a screen-shot of the results when you have finished, so please make sure the game is running in "windowed" mode before you start. If you are running in full-screen mode, please restart the game now and select "windowed" mode.

Level Tutorial

The goal of this test is to study how well a dynamic soundtrack can aid a player during gameplay. In this level, the musical soundtrack will change depending on how many enemy units are attacking you in the current wave. You will be notified on-screen when the enemies have spawned and a visual countdown timer will start. Before this timer finishes and the enemies arrive, you should listen to the soundtrack and make a prediction of how many enemies are attacking. You do this by pressing one of the five buttons at the bottom of the screen. The predictions can be: 1–2 enemies, 3–4 enemies, 5–6 enemies, 7–8 enemies or 9–10 enemies.

This level also features a further task for you to perform while you listen to the soundtrack and make your predictions. In your base you will see two Droids: a purple one and a green one, as well as corresponding spinning markers beneath them. These markers will randomly move around on the platforms from time to time and it is your task to control the Droids and keep them within their markers as much as you can. The Droids are controlled in the same way as the Hover Tanks. You can also press "1" and "2" on your keyboard to select the Purple and Green Droids respectively. Also, double-tapping the key will focus the camera on the selected unit.

Soundtrack Tutorial

You will now be played an extract of each of the musical layers that make up the soundtrack and represent the different numbers of attacking enemies. Try to listen for differences between each layer as this will help you make more accurate predictions during the test.

Example Music

Start of Test

Once you press the "Continue" button, the level will begin. You will then face ten waves of enemy Tanks. Remember to watch for the notification that tells you when the wave has spawned, and make your prediction of the number of enemies in the wave before the countdown ends and the enemies arrive.

Also remember to pay as much attention as you can to the Purple and Green Droids, and keep them within their respective markers as much as possible.

Good luck, and remember that it is not possible to perform badly. It is, after all, the music that is being tested – not you!

Thank You

Thank you for taking part in Level 2 of the *RTS Soundtrack Experiment.*

The data gathered from your participation will be invaluable in helping create a better understanding of the unique relationship between a player and a soundtrack that is present in games.

Once you press the "Fetch Results" button, you will see a print-out of the data from this test. Please take a screen-shot of this and email it to the email address below the results.

Thank you once again. The test is now complete.

Appendix 3: Level 3 Tutorial Transcript

Introduction

Greetings, Test Subject, and welcome to Level 3 of the *RTS Soundtrack Experiment*. In this final level we will be testing one last potential way in which a dynamic soundtrack can communicate with a player.

Level 3 will continue in a similar vein to the previous two levels. Therefore, if you partook in either of these, then most of the mechanics will be familiar to you. Don't worry if you didn't participate in the first two levels, however, as the following instructions will tell you all you need to know.

In this level, waves of enemies will periodically spawn to the south and attack your base. The Automatic Flame Turrets placed around the perimeter will take care of most of the enemies. If the turrets require backup, you can use the Hover Tanks provided. These are controlled using your mouse: *left-click* to select one, and *right-click* to move it or attack an enemy. You can also create a selection box by *left-clicking* and dragging the mouse; this can be used to select multiple units.

Please note: the test requires you take a screen-shot of the results when you have finished, so please make sure the game is running in "windowed" mode before you start. If you are running in full-screen mode, please restart the game now and select "windowed" mode.

Level Tutorial

The goal of this test is to study how well a dynamic soundtrack can aid a player during gameplay. In this level, the musical soundtrack will change depending on the types and quantities of enemies that are attacking your base in the current wave. You will be notified on-screen when the enemies have spawned and a visual countdown timer will start. Before this timer finishes and the enemies arrive, you should listen to the soundtrack and make a prediction of the types and quantities of enemies that you think are attacking. You make your prediction by pressing the toggle buttons at the bottom of the screen and then confirming your selection by pressing the submit predictions button.

There are three types of enemy: Yellow, Red and Brown. And each of these will attack in either a small group (2) or large group (4).

In order to attempt to simulate the typical depth of gameplay one might expect in an RTS game, this level also features two further tasks for you to perform while you listen to the soundtrack and make your predictions. In your base you will see two Droids: a purple one and a green one, as well as corresponding spinning markers beneath them. These markers will randomly move around on the platforms from time to time and it is your task to control the Droids and keep them within their markers as much as you can. The Droids are controlled in the same way as the Hover Tanks. You can also press “1” and “2” on your keyboard to select the Purple and Green Droids respectively. Also, double-tapping the key will focus the camera on the selected unit.

As well as controlling the Droids, you will also need to collect crystals to spend on reinforcing your base. When the level begins, your workers will automatically start to gather crystals from the deposits above your base. It is your task to spend those crystals every time 100 have been collected. You can see how many crystals have been collected so far by looking at the counter in the top right-hand corner of the screen.

Every time you spend 100 crystals you will either repair one of your Flame Turrets if any have been destroyed, or else you will receive a new Hover Tank.

Soundtrack Tutorial

The soundtrack of this level is made up of a number of musical layers. Three of these layers are representative of the three types of enemies that will be attacking your base: Yellow, Red and Brown. These particular layers will be added and subtracted to the music depending on whether those types of enemies are attacking in the current wave. Also note that these layers will be played at different volumes depending on the size of the group. Quiet for a small group and loud for a large group.

Press the buttons below to hear extracts of these musical layers. The last button represents an example of how these layers will sound when heard with the rest of the music.

Start of Test

Once you press the "Continue" button, the level will begin. You will then face six waves of enemy Tanks. Remember to watch for the notification that tells you when the wave has spawned, and make your prediction of the type and quantity of enemies in the wave before the countdown ends and the enemies arrive.

Also remember to pay as much attention as you can to the Purple and Green Droids, and keep them within their respective markers as much as possible and spend your crystals as soon as they reach 100.

Good luck, and remember that it is not possible to perform badly. It is, after all, the music that is being tested – not you!

Thank You

Thank you for taking part in Level 3 of the *RTS Soundtrack Experiment.*

The data gathered from your participation will be invaluable in helping create a better understanding of the unique relationship between a player and a soundtrack that is present in games.

Once you press the "Fetch Results" button, you will see a print-out of the data from this test. Please take a screen-shot of this and email it to the email address below the results.

Thank you once again. The test is now complete.

Bibliography

Adams, E. and Rollings, A. (2007) *Fundamentals of Game Design*. New Jersey: Pearson Education.

Belinkie, M. (1999) "Video Game Music: Not Just Kid Stuff" (available at www.vgmusic.com/vgpaper.shtml; accessed 2 January 2011).

Bernstein, D. (1997) "Creating an Interactive Audio Environment", *Gamasutra* (available at www.gamasutra.com/view/feature/3238/creating_an_interactive_audio_.php; accessed 20 February 2014).

Brandon, A. (1998) "Interactive Music: Merging Quality with Effectiveness", *Gamasutra* (available at www.gamasutra.com/view/feature/131670/interactive_music_merging_quality_.php; accessed 21 February 2014).

Cerqueira, M., Salazar, S. and Wang, G. (2013) "SoundCraft: Transducing *StarCraft 2*", *NIME'*13, 27–30 May 2013, KAIST, Daejeon, Korea (available at https://ccrma.stanford.edu/groups/mcd/publish/files/2013-nime-sc.pdf; accessed 1 March 2014).

Chion, M. (1994) *Audio-vision: Sound on Screen*, trans. C. Gorbman. New York: Columbia University Press.

Collins, K. (2005) "From Bits to Hits: Video Games Music Changes its Tune", *Film International* 12: 4–19 (available at http://cs.au.dk/~dsound/DigitalAudio.dir/Papers/bits2hits.pdf; accessed 11 January 2015).

Collins, K. (2007) "An Introduction to the Participatory and Non-linear Aspects of Video Games Audio", in S. Hawkins and J. Richardson (eds), *Essays on Sound and Vision*, pp. 263–298. Helsinki: Helsinki University Press (available at www.gamessound.com/texts/interactive.pdf; accessed 5 March 2014).

Collins, K. (ed.) (2008a) *From Pac-Man to Pop Music: Interactive Audio in Games and New Media*. Aldershot: Ashgate.

Collins, K. (2008b) *Game Sound: An Introduction to the History, Theory, and Practice of Video Game Music and Sound Design*. Cambridge, MA: MIT Press.

Costikyan, G. (2001) "Where Stories End and Games Begin," pp. 1–9 (available at http://cumincad.architexturez.net/system/files/pdf/b8bc.content.pdf; accessed 4 November 2013).

Costikyan, G. (2002) "I Have No Words and I Must Design: Toward a Critical Vocabulary for Games", in F. Mäyrä (ed.), *Proceedings of Computer Games and Digital Cultures Conference*, pp. 9–33. Tampere: Tampere University Press.

Cunningham, S., Grout, V. and Hebblewhite, R. (2006) "Computer Game Audio: The Unappreciated Scholar of the Half-Life Generation", in *Proceedings of the Audio Mostly Conference, 11–12 October 2006*, pp. 9–14. Piteå, Sweden: Interactive Institute.

Droumeva, M. (2011) "An Acoustic Communication Framework for Game Sound: Fidelity, Verisimilitude, Ecology", in M. Grimshaw (ed.), *Game Sound Technology and Player Interaction: Concepts and Developments*, pp. 131–152. Hershey, PA: Information Science Reference.

Ekman, I. (2005) "Meaningful Noise: Understanding Sound Effects in Computer Games", in *DAC 2005* (available at www.uta.fi/~ie60766/work/DAC2005_Ekman.pdf; accessed 12 September 2012).

Frasca, G. (1999) "Ludology Meets Narratology: Similitude and Differences between (Video) Games and Narrative" (available at www.ludology.org/articles/ludology.htm; accessed 13 February 2014).

Frasca, G. (2003) "Simulation versus Narrative: Introduction to Ludology", in M. J. P. Wolf and B. Perron (eds), *The Video Game Theory Reader*, pp. 221–235. New York: Routledge.

Friberg, J. and Gärdenfors, D. (2004) "Audio Games: New Perspectives on Game Audio", *Advances in Computer Entertainment Technology '04*, 3–5 June. Singapore: ACM (available at http://audiogames.hku.nl/---------ag/articles/Friberg_G%E4rdenfors_ACE2004.pdf; accessed 19 January 2015).

Garcia, J. M. (2006) "From Heartland Values to Killing Prostitutes: An Overview of Sound in the Video Game Grand Theft Auto Liberty City Stories", in *Proceedings of the Audio Mostly Conference, 11–12 October 2006*, pp. 22–25. Piteå, Sweden: Interactive Institute.

van Geelen, T. (2008) "Realizing Groundbreaking Adaptive Music", in K. Collins (ed.), *From Pac-Man to Pop Music: Interactive Audio in Games and New Media*, pp. 93–102. Aldershot: Ashgate.

Grimshaw, M. (2007) "The Acoustic Ecology of the First-person Shooter", unpublished PhD thesis, University of Waikato, New Zealand.

Grimshaw, M. (ed.) (2011) *Game Sound Technology and Player Interaction: Concepts and Developments*. Hershey, PA: Information Science Reference.

Grimshaw, M. and Schott, G. (2007) "Situating Gaming as a Sonic Experience: The Acoustic Ecology of First-Person Shooters", in *Proceedings of the 2007 DiGRA Conference*, pp. 474–481 (available at www.digra.org/wp-content/uploads/digital-library/07311.06195.pdf; accessed 18 October 2014).

Heeter, C. and Gomes, P. (1992) "It's Time for Hypermedia to Move to Talking Pictures", *Journal of Educational Multimedia and Hypermedia*, Winter (available at http://commtechlab.msu.edu/publications/files/talking.html; accessed 19 January 2015).

Herber, N. (2008) "The Composition-instrument: Emergence, Improvisation and Interaction in Games and New Media", in K. Collins (ed.), *From Pac-Man to Pop Music: Interactive Audio in Games and New Media*, pp. 103–23. Aldershot: Ashgate.

Hermann, T., Hunt, A. and Neuhoff, J. G. (eds) (2011) *The Sonification Handbook*. Berlin: Logos.

Hoffert, P. (2007) *Music for New Media: Composing for Videogames, Web sites, Presentations, and other Interactive Media*. Boston, MA: Berklee Press.

Jørgensen, K. (2006) "On the Functional Aspects of Computer Game Audio", in *Proceedings of the Audio Mostly Conference, 11–12 October 2006*, pp. 48–52. Piteå, Sweden: Interactive Institute.

Jørgensen, K. (2008a) "Audio and Gameplay: An Analysis of PvP Battlegrounds in World of Warcraft", *Game Studies*, 8:2 (available at http://gamestudies.org/0802/articles/jorgensen; accessed 12 September 2012).

Jørgensen, K. (2008b) "Left in the Dark: Playing Computer Games with the Sound Turned Off", in K. Collins (ed.), *From Pac-Man to Pop Music: Interactive Audio in Games and New Media*, pp. 163–76. Aldershot: Ashgate.

Jørgensen, K. (2009) *A Comprehensive Study of Sound in Computer Games: How Audio Affects Player Action*. Lewiston, New York: The Edward Mellen Press.

Jørgensen, K. (2011) "Time for New Terminology? Diegetic and Non-Diegetic Sounds in Computer Games Revisited", in M. Grimshaw (ed.), *Game Sound Technology and Player Interaction: Concepts and Developments*, pp. 78–97. Hershey, PA: Information Science Reference.

Juul, J. (2001) "Games Telling Stories? A Brief Note on Games and Narratives", *Game Studies*, 1 (1) (available at www.gamestudies.org/0101/juul-gts/; accessed 29 April 2014).

Juul, J. (2004) "The Definitive History of Games and Stories, Ludology and Narratology", *The Ludologist* (available at www.jesperjuul.dk/ludologist/index.php?m=200402; accessed 12 February 2014).

Juul, J. and Norton, M. (2009) "Easy to Use and Incredibly Difficult: On the Mythical Border between Interface and Gameplay", in *Proceedings of the 4th International Conference on Foundations of Digital Games*, ACM, 107–112 (available at www.jesperjuul.net/text/easydifficult/; accessed 13 January 2014).

Kaae, J. (2008) "Theoretical Approaches to Composing Dynamic Music for Video Games", in K. Collins (ed.), *From Pac-Man to Pop Music: Interactive Audio in Games and New Media*, pp. 75–92. Aldershot: Ashgate.

Kramer, G. (ed.) (1994) *Auditory Display: Sonification, Audification, and Auditory Interfaces*. Reading, MA: Addison-Wesley.

Kramer, G., Walker, B., Bonebright, T., Cook, P., Flowers, J. and Miner, N., et al. (n.d.) *Sonification Report: Status of the Field and Research Agenda* (www.icad.org/websiteV2.0/References/nsf.html; accessed 19 January 2015).

Lantz, F. and Zimmerman, E. (1999) "Rules, Play and Culture: Towards an Aesthetic of Games", *Merge Magazine* (available at www.ericzimmerman.com/texts/RulesPlayCulture.html; accessed 14 February 2014).

Leonard, S. (2001) "Scores of Glory, Fantasy and Plumbing: The Concise History and Principal Genres of Video Game Music' (available at http://66.165.155.21/iSphere/scores.htm; accessed 19 January 2015).

Marks, A. (2009) *The Complete Guide to Game Audio: For Composers, Musicians, Sound Designers and Game Developers*. 2nd edn. Oxford: Focal Press.

Miller, M. (1997) "Producing Interactive Audio: Thoughts, Tools, and Techniques" *Gamasutra* (available at www.gamasutra.com/view/feature/131645/producing_interactive_audio_.php; accessed 19 January 2015).

Moffat, D. C. and Kiegler, K. (2006) "Investigating the Effects of Music on Emotions in Games", in *Proceedings of the Audio Mostly Conference, 11–12 October 2006*, pp. 37–41. Piteå, Sweden: Interactive Institute.

Morasky, M. (2014) "Music in Valve Games and Media" (video recording of talk at Steam Dev Days 2014; available at www.steamdevdays.com; accessed 4 January 2015).

Murch, W. (1998) "Dense Clarity – Clear Density", republished in *The Transom Review*: 5 (1), 2005 (available at http://transom.org/2005/walter-murch-part-1/#part-2; accessed 18 October 2014).

Oswald, D. (2012) "Non-speech Audio-semiotics: A Review and Revision of Auditory Icon and Earcon Theory", in *Proceedings of the 18th International Conference on Auditory Display, Atlanta, GA, USA, June 18–21*, pp. 36–43 (available at www.david-oswald.de/downloads/Oswald_ICAD2012.pdf; accessed 18 January 2015).

Parfit, D. (2005) "Interactive Sound, Environment, and Music Design for a 3D Immersive Video Game", unpublished MA thesis, New York University, New York (available at www.davidparfit.com/Parfit05.pdf; accessed 19 January 2015).

Perron, B. and Wolf. M. J. P. (2009) "Introduction", in B. Perron and M. J. P. Wolf (eds), *The Video Game Theory Reader 2*, pp. 1–21. New York and London: Routledge.

Phillips, W. (2014) *A Composer's Guide to Game Music*. Massachusetts: MIT Press. Kindle Edition.

Pichlmair, M. and Kayali, F. (2007) "Levels of Sound: On the Principles of Interactivity in Music Video Games", paper presented to the *Digital Games Research Association Conference "Situated Play", Tokyo, Japan* (available at www.digra.org/dl/db/07311.14286.pdf; accessed 19 January 2015).

Pidkameny, E. (2002) "Levels of Sound" (available at www.vgmusic.com/information/vgpaper2.html; accessed 4 December 2013).

Ross, R. (2001) "Interactive Music ... er, Audio", *Gamasutra*_(available at www.gamasutra.com/view/feature/131489/interactive_musicer_audio.php?page=3; accessed 21 February 2014).

Salen, K. and Zimmerman, E. (2004) *Rules of Play: Game Design Fundamentals*. Cambridge, MA: MIT Press (online version at http://gamifique.files.wordpress.com/2011/11/1-rules-of-play-game-design-fundamentals.pdf; accessed 11 January 2014).

Smalley, D. (1992) "The Listening Imagination: Listening in the Electroacoustic Era", in J. Paynter, R. Orton, et al. (eds), *Companion to Contemporary Musical Thought*, vol. 1, pp. 514–554. London: Routledge.

Stevens, R. and Raybould, D. (2011) *The Game Audio Tutorial: A Practical Guide to Sound and Music for Interactive Games*. Burlington, MA: Focal Press. Kindle Edition.

Stockburger, A. (2003) "The Game Environment from an Auditive Perspective" (available at www.stockburger.at/files/2010/04/gameinvironment_stockburger1.pdf; accessed 23 May 2014).

van Tol, R. and Huiberts, S. (2008) "IEZA: A Framework for Game Audio", *Gamasutra* (available at www.gamasutra.com/view/feature/3509/ieza_a_framework_for_game_audio.php; accessed 29 April 2014).

Tuuri, K., Mustonen, M-S. and Pirhonen, A. (2007) "Same Sound – Different Meanings: A Novel Scheme for Modes of Listening", in *Proceedings of the Audio Mostly Conference, 27–28 September 2007*, pp. 13–18. Röntgenbau, Ilmenau, Germany: Fraunhofer Institute for Digital Media Technology IDMT.

Walker, B. N. and Nees, M. A. (2011) "Theory of Sonification", in T. Hermann, A. Hunt and J. G. Neuhoff (eds), *The Sonification Handbook*, pp. 9–39. Berlin: Logos.

Weinen, K. (2007) *The Rhythm to Jump and Run: The Difference between the Function of Commissioned and Popular Music in Videogames*. Munich: GRIN Verlag.

Whalen, Z. (2004) "Play Along: An Approach to Videogame Music", *Game Studies*, 4 (1) (available at www.gamestudies.org/0401/whalen/; accessed 5 March 2014).

Whitmore, G. (2003) "Design with Music in Mind: A Guide to Adaptive Audio for Game Designers" *Gamasutra*, 29 May (available at www.gamasutra.com/resource_guide/20030528/whitmore_pfv.htm; accessed 19 November 2012).

Wilhelmsson, U. and Wallén, J. (2011) "A Combined Model for the Structuring of Computer Game Audio", in M. Grimshaw (ed.), *Game Sound Technology and Player Interaction: Concepts and Developments*, pp. 98–130. Hershey, PA: Information Science Reference.

Wingstedt, J. (2006) "REMUPP: A Tool for Investigating Musical Narrative Functions", in *Proceedings of the Audio Mostly Conference, 11–12 October 2006*, pp. 42–47. Piteå, Sweden: Interactive Institute.

Wolf, M. J. P. (ed.) (2001) *The Medium of the Video Game*. Austin, TX: University of Texas Press.

Wolf, M. J. P. and Perron, B. (2003) Introduction", in M. J. P. Wolf and B. Perron (eds), *The Video Game Theory Reader*, pp. 1–24. New York: Routledge.